



LINUX

THE COMPLETE MAGAZINE ON OPEN SOURCE

VOLUME: 06 ISSUE: 10 December 2008 116 PAGES ISSUE# 71

For You

Openmoko FreeRunner

Are You Ready to Run Free?

GNOME Do + Ubiquity

Give Rise to Spontaneous Computing

Linux Scheduler

How it Copes with CPU Advances

udev Unplugged!

And Associated Tips & Tricks

PBX in a Flash

Voxzone X100P Reviewed

Session Cookies

Management Using PHP

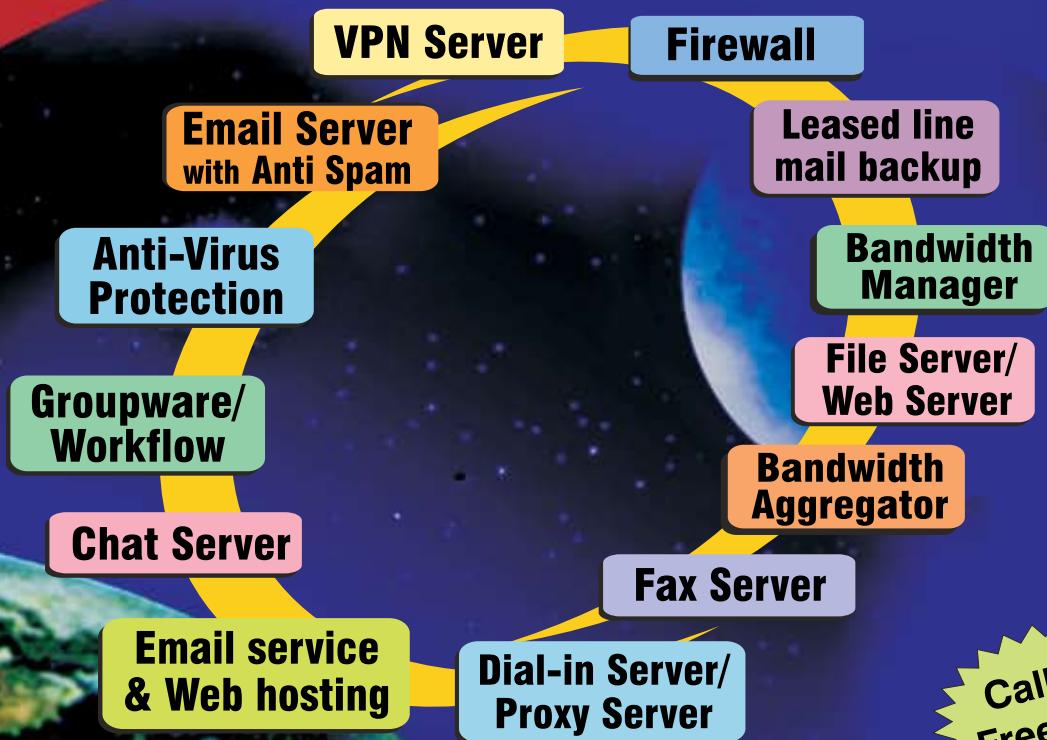


India
Singapore
Malaysia

INR 100
S\$ 9.5
MYR 19



Feature packed, complete intranet solution for the discerning enterprise/government organizations/educational and research institutions.



Call for
Free Trial

- Single SIGN ON system
- GUI based Configuration and administrative program
- Enterprise Back up and Restore functions
- Works with all kinds of Connectivity
- Multi-location support
- Migration from any existing legacy solutions
- Optional Mail/Web hosting services
- 'Optional Clustering Suite' for High availability requirement.

Carizen
SOFTWARE (P) LTD

Chennai : 044-24958222/8228/9296/0610
Mumbai : 022-66492136/37/38
New Delhi : 011 - 26239633/4/5
Bangalore : 080-22208778/9343718998
Hyderabad : 040-27551584/9391370082

Email: info@carizen.com
Web: www.carizen.com



Rainmail Intranet Server

World class network infrastructure at your fingertips

Linux Learning Centre

Pioneers in Training on Linux Technologies

Trained participants from over 36 Countries in 6 Continents

Linux OS Administration & Security Courses for Migration

LLC102: Essentials of Linux OS
LLC103: Linux System & Network Administration
LLC203: Linux Advanced Administration
LLC303: Linux System & Network Monitoring Tools
LLC403: Qmail Server Administration
LLC404: Postfix Server Administration
LLC405: Linux Firewall Solutions
LLC406: OpenLDAP Server Administration
LLC408: Samba Server Administration
LLC409: DNS Administration

Courses for Developers

LLC104: Linux Internals & Programming Essentials
LLC105: Programming with Qt
LLC106: Device Driver Programming on Linux
LLC107: Network Programming on Linux
LLC108: Bash Shell Scripting Essentials
LLC109: CVS on Linux
LLC204: MySQL on Linux
LLC205: Programming with PHP
LLC206: Programming with Perl
LLC207: Programming with Python
LLC208: PostgreSQL on Linux
LLC501: Programming with OpenGL

RHCE Certification Training

RH033: Red hat Linux Essentials
RH133: Red Hat Linux System Administration
RH253: Red Hat Linux Networking & Security Administration
RH300/301: Red Hat Rapid Track Certification Course

RHCS Certification Training

RHS333: Red Hat Enterprise Security: Network Services
RH423: Red Hat Enterprise Directory Services & Authentication
RHS429: Red Hat Enterprise SELinux Policy Administration

New Courses
RH142 & RH184



**RH142:Linux Troubleshooting
Techniques & Tools**

RH184:RHEL Virtualisation

**RHCE Exams in Bangalore
on 12 & 26 December 2008**

**RH301 from 8 Dec '08
RH333 from 6 Dec '08
RH423 from 13 Dec '08
RHS429 from 20 Dec '08**

Linux Support & Solutions
Installation, Setup and Support Solutions
for RedHat, Ubuntu, SUSE, CentOS

For more info log on to:
www.linuxlearningcentre.com
Call: 9845057731 / 9343780054
Email: info@linuxlearningcentre.com

RED HAT Training Partner
RHCE Examination Centre



Linux Learning Centre Private Limited

Registered Office & Corporate Training Centre

635, 6th Main Road, (Adj.. Bank of India) Hanumanthnagar, Bangalore 560019

Tel: +91.80.22428538 / 26600839 / 26610999 TelFax: +91.80.26600839

Cell: 9845057731 / 9343780054 Email: info@linuxlearningcentre.com

URL: www.linuxlearningcentre.com

LLC Satellite Centre - Bangalore

1291, 24th Cross, 30th Main,
BSK II Stage, Bangalore 560070

Tel: +91.80.26712928



Ready to Run Free with Openmoko Neo FreeRunner?

The Openmoko project is one of the most interesting efforts with Linux, daringly seeking to free the mobile phone from the grip of proprietary and closed-source software, and to bring the bazaar model of development to this rapidly burgeoning arena. | 22



FOR YOU & ME

- 18 GNOME Do + Ubiquity: Making Life Interactive and Spontaneous
- 22 I'm Running Free... with the Openmoko Neo FreeRunner
- 32 The Intrepid Ibex Awaits Your Command
- 34 OpenOffice.org Gets a Version Older
- 36 A Walk to Spread the Message of Freedom
- 42 Using Your Mother Tongue on the FOSS Desktop—Part I: It's Easy with KDE
- 46 When a Desi-crafted Card Meets Software... PIAF!

DEVELOPERS

- 60 Protocols to Transfer Files Between Mobiles and PCs
- 66 Let's Visit the 'Libraries'
- 70 Total Eclipse: Simplified Java Development with Ingres CAFÉ
- 74 A CAFÉ for Web Developers
- 84 How to Contribute to Open Source
- 86 Session Management Using PHP — Part 1: Cookie-based Sessions
- 92 For Aspiring Game Designers

CONTENTS

COLUMNS

- 39 FOSS is __FUN__: How To Grow the Indian FOSS Movement
- 99 The Joy of Programming: Understanding 'typedef' in C
- 100 Code Sport
- 104 A Voyage to the Kernel—Day 6: Segment 2.1

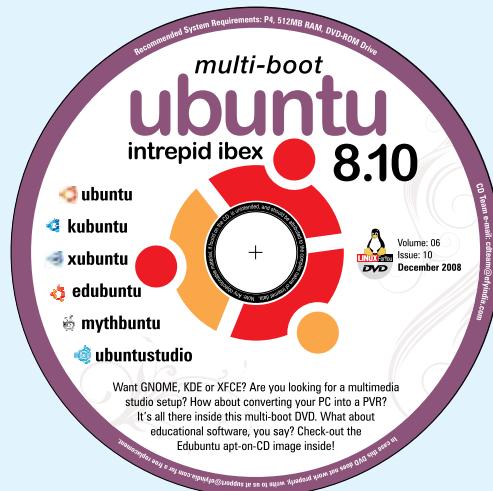
GEEKS

- 40 Put Some Colour on that Terminal!
- 50 How the Linux Scheduler Copes with Processor Architecture Advances
- 54 udev Unplugged!
- 62 Programming in Python for Friends and Relations: Part 8—Programming in Python for Mobile Gadgets Using the Web
- 81 Internationalisation and Localisation: The Tasks Ahead

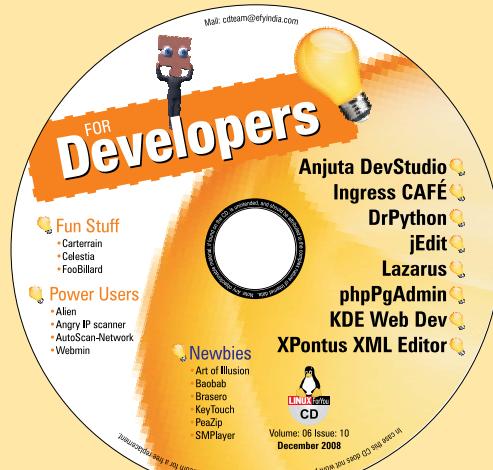
REGULAR FEATURES

- 06 Editorial
- 08 Feedback
- 10 Technology News
- 16 Q&A
- 78 Industry News
- 102 CD Page
- 108 Tips & Tricks
- 110 Linux Jobs

LFY DVD



LFY CD



NOTE All articles in this issue, except for interviews, verbatim quotes, or unless otherwise explicitly mentioned, will be released under Creative Commons Attribution-Share Alike 3.0 Unported Licence a month after the date of publication. Refer to <http://creativecommons.org/licenses/by-sa/3.0/> for a copy of the licence.

EDITORIAL

Dear readers,

This year brought about some significant releases from the FOSS ecosystem. It started with the first stable release of KDE4, what we consider to be the future of the open source desktop. Although, the initial release was tagged as a preview for developers to port their applications to, or build on top of, it left users mildly dissatisfied with the lack of tools under the hood.

What followed mid-year was the 4.1 version, and if anything has managed to produce the 'WOW' effect in the last few years, this was it. This time around, the complete software stack for desktop users was back in action, and Plasma, the desktop shell, showed many significant improvements. KDE4 was finally ready for regular desktop users to switch to. One thing I know for sure is that our in-house team is very happy, and the look and feel even won fans from amongst die-hard GNOME users.

That's about the desktop! What about the platform that many claim will replace the desktop? If you remember, we started the year with an issue about 'FOSS on Mobiles', with Openmoko creator Sean Moss-Pultz on the cover, holding a Neo 1973, the first developers' release. Mid-year saw the launch of the next edition of the Neo phones, Neo FreeRunner. This was a significant release considering that the hardware is now, finally, stable. Talking about the hardware, the Openmoko project goes a step further than the traditional FOSS-friendly vendors—the team members decided to make an open phone with even the hardware open. Likewise, the company published the CAD files under a CC licence, enabling designers to freely tailor the phone to their needs.

And, what's more, we in India were able to get a sneak peak as soon as the phone was released. This was even before it was officially available in the US. How on earth did we manage that? Well, the credit goes to IDA Systems. To know more about that, turn to Page 26. On second thoughts....you should turn to Page 22 before going to 26, because our mega-feature on OpenMoko starts from Page 22.

We hope that this 9-page exclusive on OpenMoko will not only provide you with an in-depth understanding of a true 'open phone', but will also inspire the geek in you to hack the phone, and make it even better.

Last month, we had talked about distributing Ubuntu 8.10 through the LFY DVD. However, we decided to go one step ahead and bundle it with all the other Ubuntu derivatives as well. Thanks to Niraj Sahay, a key member of the LFY Labs, we have for you a multi-boot LFY DVD, which is not only loaded with the most-requested Ubuntu and Kubuntu variants, but also with Xubuntu, Ubuntu Studio and Mythubuntu.

Wait, there's more. Navigate to the Edubuntu directory in the DVD and burn the ISO image into a CD. Once you insert the newly-burned CD into the CD drive while working on Ubuntu, it will launch a pop-up to install the add-on applications—there's no need to hack the sources files of your package manager.

I'll let you go now ...to enjoy the articles in the magazine.

Wish you all a Merry Christmas and a Happy New Year,



Rahul Chopra,
Editor, LFY
lfyedit@efyindia.com

Editor

RAHUL CHOPRA

Editorial, Subscriptions & Advertising

DELHI (HQ)

D-87/1, Okhla Industrial Area,
Phase I, New Delhi 110020
Phone: (011) 26810602, 26810603
Fax: 26817563
E-mail: info@efyindia.com

BANGALORE

No. 9, 17th Main,
1st Cross, HAL II Stage,
Indiranagar, Bangalore 560008
Ph: (080) 25260023; Fax: 25260394
E-mail: efyblr@efyindia.com

CHENNAI

M. Nackeeraan
DBS House, 31A, Cathedral Garden Road
Near Palmgroove Hotel, Chennai 600034
Ph: 044-28275191; Mobile: 09962502404
E-mail: efychn2@efyindia.com

Customer Care

E-MAIL: support@efyindia.com

Back Issues

Kits 'n' Spares
D-88/5, Okhla Industrial Area,
Phase I, New Delhi 110020
Phone: (011) 32975879, 26371661-2
E-mail: kits@efyindia.com
Website: www.kitsnspares.com

Advertising

KOLKATA
D.C. Mehra
Ph: (033) 22294788
Telefax: 22650094
E-mail: efykal@efyindia.com
Mobile: 09432422932

MUMBAI

Flory D'Souza
Ph: (022) 24950047, 24928520; Fax: 24954278
E-mail: efyum@efyindia.com

PUNE

Zakir Shaikh
Mobile: 09372407753
E-mail: efpune@efyindia.com

HYDERABAD

P.S. Muralidharan
Ph: 09849962660
E-mail: efyhyd@efyindia.com

Exclusive News-stand Distributor (India)

INDIA BOOK HOUSE PVT LTD
Arch No. 30, below Mahalaxmi Bridge, Mahalaxmi,
Mumbai - 400034 Tel: 24942538, 24925651,
24927383 Fax: 24950392
E-mail: info@ibhw.com

Printed, published and owned by Ramesh Chopra. Printed at Ratna Offset, C-101, DDA Shed, Okhla Industrial Area, Phase I, New Delhi 110020, on 28th of the previous month, and published from D-87/1, Okhla Industrial Area, Phase I, New Delhi 110020. Copyright © 2008. All articles in this issue, except for interviews, verbatim quotes, or unless otherwise explicitly mentioned, will be released under Creative Commons Attribution-Share Alike 3.0 Unported License a month after the date of publication. Refer to <http://creativecommons.org/licenses/by-sa/3.0/> for a copy of the licence. Although every effort is made to ensure accuracy, no responsibility whatsoever is taken for any loss due to publishing errors. Articles that cannot be used are returned to the authors if accompanied by a self-addressed and sufficiently stamped envelope. But no responsibility is taken for any loss or delay in returning the material. Disputes, if any, will be settled in a New Delhi court only.

Experience the
unique innovative
solution

emeric
email2sms

Mail on Mobile

Features

- Receive emails via SMS
- No GPRS required
- No software installation required
- Works with any email service: corporate, POP3 and IMAP accounts.
- Compose, forward & reply immediately
- Receive emails from multiple mailboxes
- Set your own time for receiving emails: DND
- Set filters to receive emails from selected senders
- View attachments without add-on mobile applications



netCORE
The Innovation Company

sales@netcore.co.in | www.m3m.in

MUMBAI | NEW DELHI | CHENNAI | BANGALORE | HYDERABAD | PUNE | AHMEDABAD | KOLKATA
9820032324 | 9350101237 | 9380388685 | 9341609111 | 9392224001 | 9373727934 | 9824192328 | 9331420404



You said it...

 LFY's November 2008 issue was really great (and different from other issues), as it helped Windows users switch to FOSS. The problem that most people face when changing an OS, is the absence of applications they used in the previous OS—the familiar GUI, and some typical features.

The LFYDVD team did an excellent job by compiling the ultimate software DVD—it was very well categorised, covering all sections. And most of the software are replacements for common Windows software—in fact, with added features. Though I cannot use the software myself (in my Fedora 8 x86_64), it will really help me to distribute FOSS as a replacement for proprietary software.

The wget article was great. In the portable applications article, the author missed a great portable tool—YamiPod [www.yamipod.com]. It is an iPod manager, which needs no installation, can transfer music to and from the iPod, anywhere. Just keep it on your iPod, and transfer songs anywhere by launching it from within the iPod. The best part is it is available for Windows, Linux and MacOS, and keeps updating.

I will be waiting for Fedora 10 and more surprises from the LFY team in forthcoming issues.

—**Arjun Pakrashi, Kolkata**

ED: We're glad that you find the Windows software DVD with FOSS tools handy—it took the team a substantial amount of time to hunt for and compile all the apps in the DVD under the proper categories. Also, thanks for the tip on YamiPod—sounds very interesting! However, the article on Portable Apps was a review of the portableapps.com service that offers some of the FOSS tools as portable

apps for Windows, not really on 'portable applications' in general. Oh, and guess what? Fedora 10 will be bundled with the New Year issue. :-)

 Apart from the well-packed DVD, this month's LFY has some nice articles. I liked the one that explains GRUB in detail. The DVD packaging is good. LFY is getting more and more interesting.

—**Rony, on ILUG-Bombay mailing list**

 I read about Mohd Azwar from Malaysia asking for help on configuring 3G on Linux in this month's [November 2008] 'You said it' section. He can easily configure 3G in Mandriva from the Mandriva Control Centre, and the new Intrepid has better 3G support, along with lots of regression.

—**Shashwat Pant, by e-mail**

ED: Thanks for the information! I hope Azwar is able to configure his system.

Errata

Misprints in November 2008 issue:

- Pg 4: Google Chrome was misprinted as Goole Chrome.
- Pg 48: The version numbers for Ubuntu Hardy and Intrepid were mentioned as 7.04 and 7.10, respectively. They should be 8.04 and 8.10, respectively.
- Pg 58: In the first line of second code snippet `http://www.` was printed twice. The correct entry should be `wget http://wwwpendrivelinux.com/`
- Pg 59: For the code snippet of point number 5, it should be `wget` and not `get`.
- Pg 76: In the second code snippet of column 2, `grub> setup (hd0)` was printed twice.

And now it's time to talk about an exceptional achievement!

 **11-year-old Indian becomes a Red Hat Certified**

Engineer—makes a great headline, doesn't it, considering it is supposed to be one of the toughest examinations to crack? In fact, what makes it more sensational is that this boy is the youngest candidate in the world.



Born on October 21, 1997, M. Kiranraj is a 6th grader at St. Joseph Hr. Sec. School, Poonamallee. He got interested in computers a couple of years back. Having seen his interest, his parents—P Mohanavelu, who works as a programmer at Sri Venkateswara College of Engineering in Sriperumbudur, and Lathashree, a home maker—encouraged him towards taking this examination. He was coached by his father, who himself is a RHCE, for just two months. Kiranraj is good at studies and always tops his class. He wants to develop his skills in computer engineering.

Another interesting fact is that he secured 100 per cent in the first part of the test (troubleshooting). He scored 71 (RHCT components) and 82.10 (RHCE components) in the subsequent parts of the examination, totalling 253.10 out of 300, to become the "Youngest Red Hat Certified Engineer".

Don't forget to tune in next month to get up close and personal with the boy genius!

Please send your comments or suggestions to:

The Editor

LINUX FOR YOU Magazine

D-87/1, Okhla Industrial Area, Phase I,

New Delhi 110020; Phone: 011-

26810601/02/03;

Fax: 26817563; Email: lfyedit@efyindia.com;

Website: www.OpenITis.com



Retain & Retrieve your mission critical emails for several years!!!

emeric

MailArchiva

Email Archiving & Compliance solution

- ✉ Compliance (SOX, HIPPA)
- ✉ Insurance against Litigation
- ✉ Policy, Security & Retention rules

- ✉ Indexing Content & Attachment
- ✉ Encrypted Backup
- ✉ Compression & De-Duplication

Key Benefits

- Seamlessly integrates with any mailing solution (Exchange, Domino, Linux)
- Automated Archiving without end user or administrator intervention
- Quick, Easy Search & Retrieval
- Back up system for lost emails
- No special database required

Partial Client List

- Geojit Financial Services Ltd.
- IDBI Capital Ltd.
- Apollo Sindhoori Capital Investment Ltd.
- Flat Products Equipment (I) Pvt. Ltd.
- IndusInd Bank Ltd

netCORE
The Innovation Company

sales@netcore.co.in | www.netcore.co.in/ema

MUMBAI | NEW DELHI | CHENNAI | BANGALORE | HYDERABAD | PUNE | AHMEDABAD | KOLKATA
9820032324 | 9350101237 | 9380388685 | 9341609111 | 9392224001 | 9373727934 | 9824192328 | 9331420404

TECHNOLOGY NEWS

Play H.264 video on TI SoC

Texas Instruments has announced a digital media processor based on the DaVinci technology, the TMS320DM357. The DM357 is a low-cost, ARM-based processor that includes a royalty-free H.264 codec at D1 resolution for video compression, as well as MPEG-4, JPEG and G.711 codecs that do not require licensing fees or royalties to TI, and an integrated Ethernet Media Access Controller (EMAC) to help developers reduce their bill of material (BOM) costs.

The DM357 processor includes an ARM926EJ-S core that runs at 270 MHz, as well as a co-processor to speed H.264, MPEG-4 and JPEG (HMJCP) processing, in addition to an integrated video processing subsystem. The DM357 processor and DVEVM take advantage of all the tools and support included in the DaVinci technology portfolio to help OEMs save months of time. The application programming interfaces (APIs) common across DaVinci offerings also means that developers familiar with DaVinci technology or ARM development can quickly begin creation of their products with virtually no learning curve.

When coupled with the DVEVM, developers are able to get started immediately with product development. The DVEVM helps them achieve the fastest possible time to market, with optimised MontaVista Linux, a uboot loader and drivers for the complete peripheral set. Rounding out the DVEVM are the H.264, JPEG, MPEG-4 SP and G.711 codecs, plus Video Input/Output, Audio In/Out, an external EMAC, USB 2.0 On-The-Go and JTAG for test.

The TMS320DM357ZWT processor is now open for order entry from TI and TI authorised distributors. It is priced at US\$ 21.22 in 100 unit volume. The highly integrated device is packaged in a 16 x 16, 0.8 mm pitch ball grid array package. The TMDSEVM357 Digital Video Evaluation Module is also now open for order entry at a cost of US\$ 895. Codecs will be available for download in mid-December. For more information, please visit www.ti.com/dm357pr.

A separation kernel and embedded hypervisor

Highlighting its continued leadership in aerospace and defence software, LynxWorks has announced the availability of LynxSecure 2.0, a separation kernel and embedded hypervisor for high-assurance systems.

Traditional systems require a separate processor and system, one for each deployed OS environment and supported applications. The ability of LynxSecure to consolidate heterogeneous OS environments enables developers to engage a diverse array of applications on a single processor, which reduces hardware costs and allows for easier reuse of legacy software. In addition, LynxSecure supports a lightweight Application run-time environment that can be used for creating secure applications without an intervening OS, which can be evaluated to the required assurance level up to EAL-7. With its extremely small code size, LynxSecure maintains hard real-time characteristics and determinism for real-time applications. The software is the first separation kernel and hypervisor to bring multi-core processor support to the high assurance world. For more details, visit www.lynxworks.com



Mainline kernel introduces support for Atmel's MPU

Atmel announced the availability of the latest Linux mainline release, v2.6.27, for its 400 MHz ARM926EJ-S-based AT91SAM9G20 embedded microprocessor, and for other members of the AT91SAM9 family. A Linux distribution based on Linux v2.6.27 is available from Atmel's AT91SAM Linux portal at www.linux4sam.org. It includes the complete Linux v2.6.27 kernel, the Linux patch for the AT91SAM9G20-EK, device drivers, pre-built demonstrations and the Angstrom/OpenEmbedded building environment.

Complementary products and support are available through TimeSys, including an embedded Linux ReadyKit for the entire AT91SAM9 series, which includes the AT91SAM9G20. The ReadyKit comprises a pre-built Linux kernel, device drivers, a GNU-based cross toolchain, a *glibc*-based root filesystem complete with selected development libraries, 14 days of technical support and access to a wide range of support documentation.

The 400 MHz AT91SAM9G20 features Atmel's DMA (direct memory access) and distributed memory architecture that, together with the 6-layer bus matrix, enables multiple simultaneous data transfers between memories, peripherals and external interfaces without consuming CPU clock cycles. The external bus interface (EBI) is clocked at 133 MHz for high-speed transfers to off-chip memories. This architecture gives the device the high internal and external data bandwidth required by many embedded networked applications. The AT91SAM9G20-EK kit is available at a unit price of \$500 with a free Linux BSP. More information at www.atmel.com/products/at91/default.asp

Large **SMP** systems are now as affordable as **HPC**

Problem Statement	Relative Cost		
	SMP	x86 clusters	Tyrone SMP
R&D and manufacturing cost	High	Low	Low
Initial acquisition cost			
Installation and ongoing cost	Low	High	Low
Programming cost			

High Performance Computing (HPC) applications have always needed faster and larger systems, historically served by SMP systems. An alternative for large SMP systems evolved in the late 90s when x86 processors exceeded the performance of RISC processors, promoting the adoption of commodity server clusters connected with high-speed interconnects. Cluster solutions deliver more performance than traditional SMPs, but are more difficult to program, implement and manage.

We at Tyrone systems deliver SMP system, a software based computing-architecture that combines the advantages of shared-memory systems and the price points of commodity clusters by leveraging off-the-shelf x86 components

20-CPU/80-core SMP system with 160GB RAM @ US\$70,000/-*
(20 x Intel Xeon Quad-Core 2.5GHz, 160GB RAM, 20 x 146GB 10000RPM SAS HDDs in 7U rack-mountable)



*Conditions apply

Please do write to us or call us for further details.

Netweb
TECHNOLOGIES
ISO 9002:2000 Company

Corporate Office

B-1/ A25, 2nd Floor
Mohan Co-Operative Industrial Estate,
Mathura Road, New Delhi – 110044

Bangalore : +91 80 41146565/22861407
Chennai : +91 44 43531009/26631975
Kolkata : +91 33 23232055/23232056

Website : www.netwebindia.com
www.tyronesystems.com
Email : info@netwebindia.com
netweb@vsnl.com

TECHNOLOGY NEWS

Linux-based wrist PC

Parvus Corporation has announced the new Zypad WR1100 rugged wrist-worn personal computer designed for harsh field conditions, and claims to be an ideal solution for military, security, and emergency service field and in-vehicle applications. This x86 compatible rugged wearable computer can be worn comfortably on the user's wrist for hands-free operation.

The WR1100 can be quickly configured to access any remote host system through its integrated wired and/or wireless interfaces using its Linux OS. The unit integrates a number of innovative features, including 802.11 and Bluetooth/Zigbee interfaces, a GPS receiver, electronic compass, biometric fingerprint sensor, and a tilt-and-dead reckoning system that detects the position of the user's arm and sets the system to standby mode when the arm is hanging down beside the body.

The Zypad WR1100 is available from stock to 12 weeks leadtime. More information can be found at www.parvus.com/products/MilitaryAerospace/WearableComputers/ZypadWR1100



Flash goes 64-bit, finally!

Adobe has released an alpha version of the 64-bit Adobe Flash Player 10. Till now, users who ran 64-bit Linux and needed Flash support had to depend on 32-bit emulation to run the 32-bit version of Flash on a 64-bit OS. The alpha release brings in native installation on 64-bit Linux distributions for the first time.

Flash Player 10 introduces new expressive features and visual performance improvements that allow interactive designers and developers to build immersive Web experiences. You can download it from labs.adobe.com/downloads/flashplayer10.html and learn more by reading the FAQ at labs.adobe.com/technologies/flashplayer10/faq.html

USB 3.0 specification now available

The USB 3.0 Promoter Group has announced the completion of the USB 3.0 specification, the technical map for device manufacturers to deliver SuperSpeed USB technology to the market. Claimed to bring in significant power and performance enhancements—with data transfer rates up to ten times faster compared to USB 2.0—SuperSpeed USB promises backward compatibility with billions of USB-enabled PCs and peripheral devices currently in use by consumers.

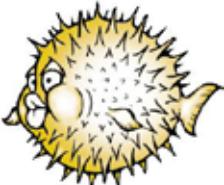
It is anticipated that initial SuperSpeed USB discrete controllers will appear in the second half of 2009 and consumer products will appear in 2010, with adoption continuing throughout 2010. The first SuperSpeed USB devices will likely include data-storage devices such as flash drives, external hard drives, digital music players and digital cameras. GPL'd Linux drivers based on the specs are apparently also being developed by MCCI Corp. and Synopsis. For more information about the USB 3.0 specification, visit www.usb.org/developers



OpenBSD 4.4 for the security conscious

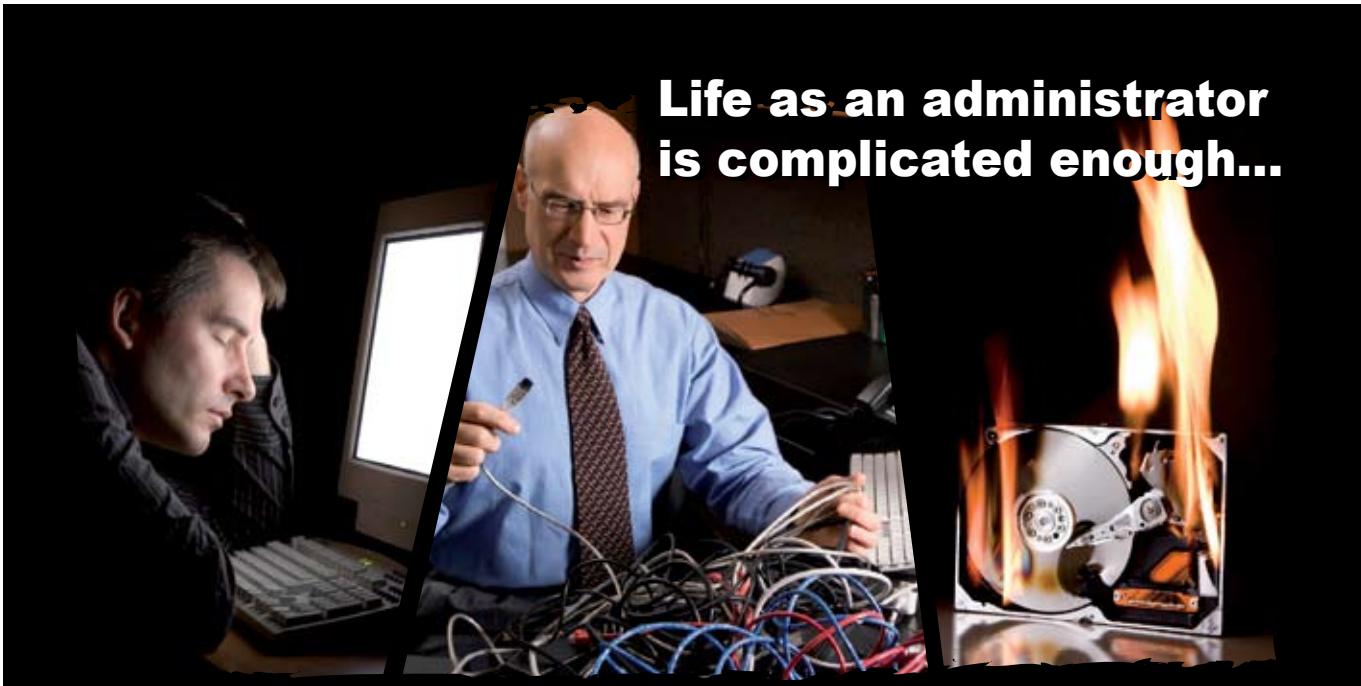
OpenBSD, a BSD OS with a strict focus on security and advanced security features, has got a version upgrade on November 1, with the release of OpenBSD 4.4. Announcing the release, project lead Theo de Raadt wrote: "This is our 24th release on CD-ROM (and 25th via FTP). We remain proud of OpenBSD's record of more than ten years with only two remote holes in the default install." The new version provides significant improvements, including new features, in nearly all areas of the system: new or extended platforms for sparc64, socppc, landisk; improved hardware support; new tools and functionality; assorted improvements and code cleanup; install and upgrade process changes; OpenSSH 5.1; over 4,500 ports. For a detailed list of changes visit www.openbsd.org/44.html.

To download, go to www.openbsd.org/ftp.html.



NetBeans 6.5 previews PHP and Python support

The NetBeans community has announced the release of NetBeans IDE 6.5. Some of the feature highlights of the new release include: feature-rich tooling for PHP, such as syntax highlighting, code completion, code generators, debugging, database wizards, and FTP support; an editor for JavaScript development, including CSS/HTML code completion; the ability to debug client-side JavaScript code within both Firefox and Microsoft Internet Explorer browsers; enhanced support for Spring, Hibernate, Java Server Pages, Java Persistence API. Visit www.netbeans.org/downloads/index.html to select the version you want to download.



**Life as an administrator
is complicated enough...**

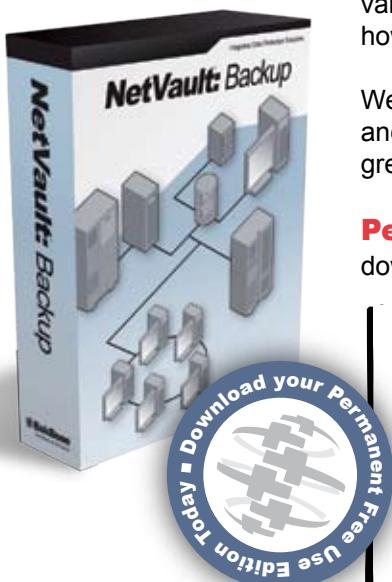
**But Backup and Recovery for your
Linux Servers does not need to be.**

NetVault: Backup simplifies backup and recovery
without compromising **Functionality** and **Scalability**

NetVault: Backup provides unmatched Data Protection for all major variants of Linux. We are now offering you the chance to see just how good NetVault is at no cost.

We provide continuous data protection (CDP) for your Linux servers and advanced application protection and recovery for MySQL, PostgreSQL, Oracle, Exchange, DB2 to name just a few.

Permanent FREE Use Edition for Linux is available for download at <http://www.bakbone.com/nvbu/redhat/>



NetVault is a true Linux data protection solution featuring:

- Online backup
- Point and click recovery
- Fully automated protection and recovery support for Linux based applications
- Virtual Tape Library (VTL) with disk staging
- SAN support with LAN free backups.
- Backup duplication for off-site storage of backups
- Support for MySQL native replication, restore to table level and to alternative databases or instances.

TECHNOLOGY NEWS

Sponsored by:



Integra Micro Systems

A Linux server the size of a pair of dice

Digi International has introduced the Digi Connect ME 9210 with Digi Embedded Linux, which is claimed to enable full Linux development in space-constrained devices. Digi Embedded Linux is the latest version of Linux optimised for development on Digi embedded modules and microcontrollers. About the size of a pair of dice, the high-performance Digi Connect ME 9210 is the smallest embedded device server available with Linux.

Digi Embedded Linux supports kernel 2.6.26. It's claimed to offer the highest level of speed and memory available on a device server with a 75 MHz ARM9 processor, 8 MB RAM and 2 or 4 MB Flash. It also features the most peripheral interfaces, including 10/100 Ethernet, serial, SPI, I2C, GPIO, CAN, 1-wire and integrated Flexible Interface Modules (FIMs). FIMs provide custom interfaces for tailoring the module to the users' exact application needs.

The Digi Connect ME 9210 features an integrated, NIST-certified AES accelerator that provides secure network communication. The accelerator provides 10 times the encryption speed of software-only solutions. It also features unique power functionality, including power-over-Ethernet (PoE) support to allow the design of devices that require no external power; and Digi Dynamic Power Control, a complete set of hardware and software features for product designs that demand low-power consumption and advanced power management. This reduces the power required to operate the Digi Connect ME 9210 and saves power on PoE networks.

Based on Digi's recently introduced NS9210 ARM9 microprocessor, the Digi Connect ME 9210 will support the extended lifecycle of embedded products. The NET+OS operating system also includes support for advanced secure networking protocols such as IPv6, SNMPv3 and SSL, further supporting long-term usability. For more information, visit www.digiembedded.com/me9210



Yellow Dog 6.1 refreshes package base

Yellow Dog Linux 6.1, targeted to Apple G4/G5, Sony Playstation 3, PowerStation, and IBM Power Systems, has been released on November 19, 2008 offering several end-user and development tool improvements over the previous version. As always, Yellow Dog Linux is also built upon the CentOS foundation, a derivative of Red Hat Enterprise Linux.

According to the media release, Yellow Dog Linux v6.1 comes with Firefox 3.0 and OpenOffice.org 2.3 (v3.0 coming to YDL.net Enhanced soon), a vastly improved graphical wireless configuration tool, and the introduction of ps3vram functionality, which enables use of PS3 video RAM for temporary storage or swap, in addition to the latest kernel 2.6.27. For developers, v6.1 offers GCC 4.1.2, the open portion of the IBM Cell SDK v3.1, and through a working relationship with the Barcelona Supercomputing Center, YDL v6.1 now ships with the new Cell Superscalar.

The new version is available via YDL.net Enhanced accounts purchased at the Fixstars Store. The public mirrors will offer v6.1 downloads approximately by December 19, 2008.

Debian Lenny update: RC1 for installer released

The month September came and went away, but Debian Lenny (v5.0) that will replace Etch (v4.0) as the new stable release was nowhere to be seen. The news is that Lenny won't come out any time before the first quarter of the next year, or even later. However, the news is not all that bad as it looks like the distribution is finally headed towards that 'stable' release with the release of the RC1 of the Debian Lenny installer on November 12, 2008.

The release flaunts: improved support for Live-CD installation media (supposed to be faster and more reliable than earlier releases); support for some NAS devices based on Marvell's ARM-compatible Orion chip; support for hardware speech synthesis (speakup); upgrade of packages early in pkgsel—for example, to get available security updates for base system packages; support for loading firmware from (removable) media during the installation; and more... To download the latest version, you can head to www.debian.org-devel/debian-installer.

Smallest computer, now in India

Comptek has launched UMPC in India, claimed to be the smallest PC in world with all features of a tablet PC including Wi-Fi, Bluetooth and Web camera



inbuilt. Weighing only 529 grams, the Wibrain B1 Ultra Mobile Computer is a small (approximately 7.5" x 3.25"), Tablet PC with touch screen, mouse pad, 1.2 GHz processor, 1 GB RAM, 60 GB HDD, integrated stereo speakers, USB and 24 Pin connector for external monitor, battery backup and comes loaded with a version of Ubuntu Linux or Windows XP. For more information visit compteki.com.

Microsoft®

BE LOOKED UP TO

.....
Be the one to save the day for developers with a solution, or share something new to become the one everyone would talk about. Start a new project, join an existing one, or download softwares created by fellow developers.
.....

ENTER A NEW HALL OF FAME AT WWW.CODEPLEX.COM,
THE MICROSOFT OPEN SOURCE PROJECT HOSTING WEBSITE.

CodePlex



McCann/Msft/08

Q & SECTION

Q I am using Firefox 2.0 and have been facing a lot of problems with the increase in Flash content on Web pages nowadays. Is there any way in which I can restrict Flash from playing streaming media on my Web browser?

—Prakash Badal, Mohali

There is a Firefox extension available at flashblock.mozdev.org that blocks all Macromedia Flash from loading. It leaves a placeholder on the Web page that allows you to click to download and then view the Flash content. So, now you have the choice of viewing a Web page with or without Flash.

Q I use Fedora and have forgotten my root password. Is there any way I can recover or reset my root password? Please help me, as I am unable to install packages on my system.

—Nilanjan Banerjee, Kolkata

You can reset the root password by logging into single user mode, which is also called a rescue mode. To log in to the single user mode, follow the steps given below. As Fedora uses GRUB, I will give the steps only applicable to GRUB:

1. Start your computer and select the 'Fedora on GRUB' menu.
2. Now press the E key to edit the parameters.

3. Select the line starting with the word 'kernel' (generally the second line) and press the E key again to enter the edit mode.
4. Append S at the end of the line and hit the Enter key.
5. Now press the B key to boot into single user mode.

After booting you get a prompt where you need to type the following:

```
passwd
```

Now, just enter the new password you want for your root account. Reboot and you are done!

Q I have MySQL 5 installed on my system and I get an error while trying to connect to it. The error is as follows:

```
ERROR 1045 (28000): Access denied for
user 'root'@'localhost' (using password:
NO)
```

Please help! It's urgent!

—Satyaprakash, by e-mail

To get around this problem you need to reset the password. First, stop the mysql server:

```
# /etc/init.d/mysql stop
```

Now open the /etc/my.cnf file in an editor, and add the following line under the [mysqld] section:

```
skip-grant-tables
```

Now restart the server again:

```
# /etc/init.d/mysql start
```

...and execute the following command:

```
# mysql -u root mysql
```

```
mysql> UPDATE user SET Password=PASSWORD('newpassword') WHERE USER='root';
mysql> FLUSH PRIVILEGES;
```

After successfully setting the password, undo the changes to the /etc/my.cnf file and restart the server. You should be able to connect with your new password.

```
# mysql -u root -p
Enter password: newpassword
```

Q I am planning to buy a laptop. Can you provide me some information on which laptops are compatible with Linux?

—Asim, by e-mail

Nearly all brands have laptops that are compatible with Linux. Since you have not mentioned your budget or the system configuration you'd prefer, it's difficult to make suggestions. You can have a look at www.linux-laptop.net. The website has a good compatibility list. Hope this will help you decide what to buy.

Q I use Firefox 2.0 and have been facing a lot of problems with the increase in Flash content on Web pages nowadays. Is there any way by which I can restrict Flash from playing on my Web browser?

—Manas Tripathi, Gurgaon

There is a Firefox extension available at flashblock.mozdev.org that blocks all Flash content from loading. It leaves a placeholder box on the Web page that allows you to click in order to download the content and then view them if you so wish. So, now you have a choice to view a Web page with or without Flash. 



India's leading Open Source event is coming...

OPEN SOURCE INDIA: TECH DAYS

It's happening at...

CHENNAI

It's happening on...

12, 13 & 14 of MARCH 2009

It will be useful for...

SOFTWARE DEVELOPERS AND IT MANAGERS
(And Newbies too!)

And, we seek YOUR help to make it the best in Asia.

Open Source India, earlier known as LinuxAsia, is going to be held at the Chennai Trade Centre from 12th to 14th of March 2009. Being organised in a new city—Chennai, the event hopes to bring in many new developments while maintaining the strengths of the past.

We seek your support to make it the best (and the biggest) open source event in Asia. You can contribute by (a) speaking at the event, (b) joining as a volunteer or (c) just by being there as an attendee.

Powered By: _____

Shoot us an email at countmein@osiweek.com, and let us know how you can help us in building an Indian Open Source event into Asia's best.



Exhibitors and Sponsors may contact us at: sales@osiweek.com



Let's Try

For U & Me

GNOME Do + Ubiquity

Making Life Interactive and Spontaneous



Heard about a tool called Quicksilver that those Mac users rave about? About time you told them about Do and Ubiquity.

Imagine a world where you can launch any application, add songs to your playlist, locate and open any file, send e-mails and IMs, search the Web right from the desktop with a variety of search engines (Google, Wikipedia, Yahoo, Amazon, et. al), send your tweets without having to open a separate Twitter client or a browser, upload photos to Flickr, and do a whole lot more without having to leave your keyboard. Now try imagining doing all that with just a few keystrokes, from a single application! Awesome, *right?*

Well this is not a sci-fi future world that I'm asking you to imagine; this world already exists! In fact, it's been a reality to Mac users since ages in the form of Quicksilver. Now Linux users can boast of the same with GNOME Do—a powerful and speedy remote control for your GNOME desktop. (Oh, and it's powerful, speedy and sensational on other GNU/Linux desktop environments, too!)

GNOME Do started off as a university project by David Siegel. You can read more about it in the GNOME Do white paper available for download at davebsd.com/do/gnome_do_white_paper.pdf. The paper talks about related works: Quicksilver and GNOME Launch

Box, technical approaches, entity resolution, and open source methodologies, among other things. Here are a few snippets from it: "Our intent is to create an interface that takes advantage of the precision and expressiveness of the keyboard, while remaining intuitive enough to appeal to novice users..

"Judging from our interactions with users and contributors, we are fairly certain that we have around 50,000 users at this point (up from a couple of hundreds last semester). We consider GNOME Do a remarkable success for seven months into our first open source project, after just recently switching to Linux and learning to use C#, Mono and GNOME as we went along. This is a testament to the 'liveness' and receptiveness of the free software community, and the flexibility and ease of use of tools like Bazaar, Launchpad, and most notably, Mono."

Now let's delve a little further into this GNOME Do world and you can see for yourself why there is so much ado about the 'Do'.

Get 'Do'

GNOME Do is available for Debian, Fedora, openSUSE, Foresight Linux, Gentoo and all other major GNU/Linux distributions. Follow the installation instructions at the official

site [do.davebsd.com] to install it on your machine if it's not already installed on your computer, or if you have an older version—for example, openSUSE 11 comes bundled with v0.4 and new plug-ins are not compatible with this version.

Using ‘Do’

Once installed, you can invoke it with a hot key. The default key combination is *Win+Space*, but you can easily configure a different one under *Preferences*. The way it works is simple: first you search for an item and next you instruct Do what action you want to perform on it. When you press the hot key combination and bring GNOME Do to the foreground, you will find two boxes side by side (Figure 1). You can navigate between them using the *Tab* key.

The first one is the ‘item’ box. Here you type whatever it is that you are looking for, maybe document ‘XYZ’? Type in XYZ, and as you’re doing so GNOME Do will find documents with a similar name. The search is adaptive, so Do will recognise which items you are searching for, based on its previous experience.

Now, the second box is what is called the ‘Action’ box. As the name implies, here you instruct Do about what action you want to perform on the item in the search box. So, for a document or an application, you may see an option like ‘Open’. If you press the down arrow key, it will open a drop down list that will list alternate actions that can be taken (Figure 2). Depending on what action you select, you may be offered an optional third box that modifies or adds to the action. For example, if your action was ‘Open With’, in the third box you can choose which application from, say, *gedit*, *kate*, OpenOffice.org, etc. Refer to Figure 3.

People typically make the mistake of thinking Do is just a launcher. Yes, it is a launcher—much better than *Alt+F2*—but it is also much more than it. To quote from the main wiki page [do.davebsd.com/wiki/index.php?title=Main_Page]: “GNOME Do not only allows you to search for items in your desktop environment (e.g., applications, contacts, bookmarks, files, music, etc), it also allows you to specify actions to perform on search results (e.g., run, open, e-mail, chat or play). Want to send an e-mail to Mom? Simply type ‘mom <Tab> email’. Want to listen to some music? Just type ‘beatles <Tab> play’. GNOME Do provides instantaneous, action-oriented desktop search results that adapt to reflect your habits and preferences. For example, if you use the Firefox Web browser often, typing ‘f’ in Do will launch it.”

‘Do’ plug-ins

As a Quicksilver clone, this also has a plug-in architecture that allows the application to be extended with new items and actions. Do comes with a set of plug-ins like Firefox, File, etc, preinstalled, so it instantly has access to Firefox favourites, applications, user documents, etc. Right click on the Do icon (3 Gears) on the GNOME panel and click on *Preferences→Plugins*. Here you can add/configure/remove plug-ins, after selecting from official or community plug-ins (Figure 4).

Here is a list of my top plug-ins that you can start off with:

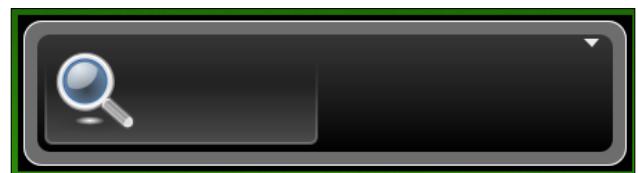


Figure 1: The GNOME Do application



Figure 2: The drop down ‘Action’ menu



Figure 3: Selecting ‘Open With’ from the ‘Action’ menu brings up a third box to choose an application



Figure 4: List of plug-ins

1. *Twitter*: If you are a Twitter addict, you will find this to be one indispensable plug-in. It provides a good and

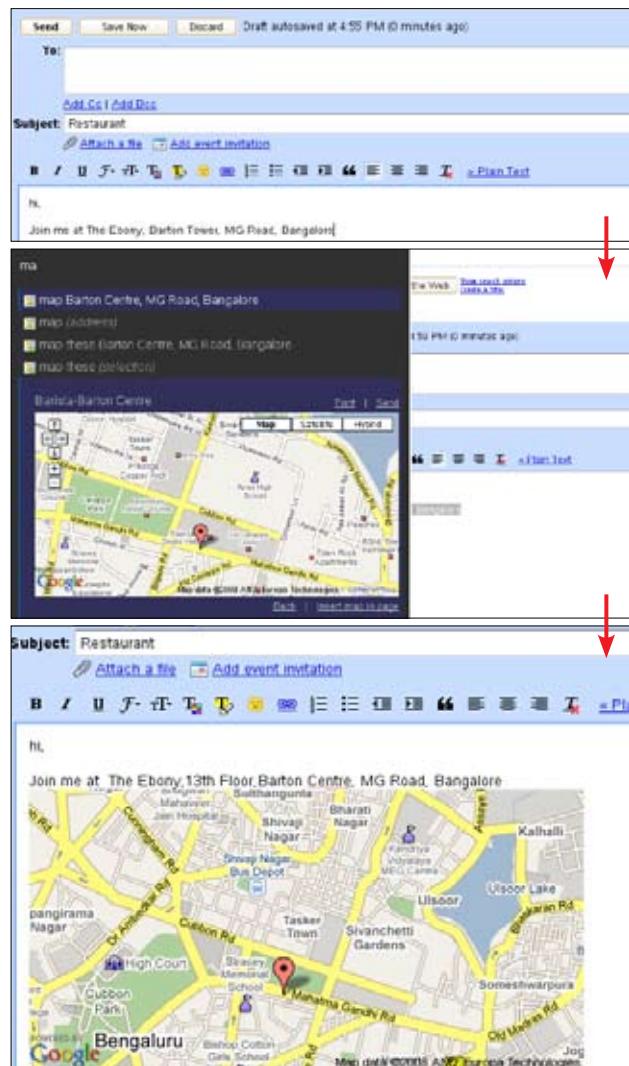


Figure 5: The interactive Ubiquity: How to insert Google Map in e-mail

quick way to update your Twitter right from Do, without having too much distraction. Apart from that, it also displays tweets from people you are following! Although I wish there was a better notification mechanism, something like Growl on Mac, and also a way to fetch the last 'X' number of tweets.

- Opensearch*: This one allows you to search right from your desktop using a variety of search engines—from Google, Yahoo, eBay, CreativeCommon, Answer.com and Amazon.com to Wikipedia. Depending on what you want to find, you can easily get the information you want from Do.
- Tasque*: This one allows you to create a new task in Tasque. For those of you who don't know what it is, Tasque is this nice little task management application (To-do list) for the Linux Desktop, which also has integrations with the Web-based task manager 'Remember the Milk'.
- Rhythmbox/Banshee/Amarok*: Depending on the player you use, install the plug-in to search, play and control music, all within Do. Though, Banshee and Amarok plug-ins have been removed for the latest version. :-)
- Flickr*: You can quickly upload one photo or a bunch of



Figure 6: The 'about:ubiquity' page

photos to Flickr without accessing your Flickr account from your browser.

- Evolution, GMail and Pidgin*: Index all your contacts and quickly e-mail or IM them right from Do.

These are just a few major plug-ins; there are a whole lot of others out there to make life at your desktop much easier and faster. You can find a list of plug-ins on the official site at do.davebsd.com/wiki/index.php?title=Category:Plugins, and you can always Google to find some more.

The available plug-ins are not as extensive as those available for Quicksilver, nor does Do support triggers, but that's expected as Quicksilver has been around for more than four years and Do is still far away from a v1.0 release. So for a 0.5 version, it's pretty good! And the good part of this is that it's a great opportunity to contribute, so if you find you want to do something, but there's no plug-in for it, go ahead and write it—it's open source!

Note: Version 0.6 is out, but it's known to have some stability issues as of now; you can read more about this on David Siegel's post at blog.davebsd.com/2008/09/16/a-cautionary-word-about-gnome-do-06.

Ubiquity—Do for Firefox

Now from GNOME Do, I'm going to jump into another related tool called Ubiquity. In fact, Ubiquity merits an article of its own, but since both are similar, it made sense to combine the two. Ubiquity, which is "an experiment into connecting the Web with language", is a product of the Mozilla Labs. It's still in the early stages of development, the latest version being 0.1.2. To quote from the Mozilla site: "Ubiquity is an experimental Firefox extension that gives you a powerful new way to interact with the Web."

Instead of instructing Firefox about where you want to go by typing Web addresses into the URL bar, you can tell Firefox "...what you want it to do" by typing commands into a new Ubiquity input box, similar to the way you do it in Do.

Just like Do, you invoke the Ubiquity input box through a hot key—the default is *Alt+Space*, and again, this is configurable. But what mainly sets it apart from Do is the

This page contains Ubiquity commands. If you'd like to subscribe to them, please click the button to the right.

[Subscribe...](#)

Figure 7: Subscribe to Ubiquity commands notification

clipboard and basic natural language support. So you can select some part in a page and issue commands like ‘e-mail this to John’ and Ubiquity will understand that by ‘this’ you meant the selected text. This is sorely lacking in GNOME Do. You will be able to better appreciate the possibilities that this context support offers after this map example:

Let's say you're arranging to meet up with a friend at a restaurant, and you want to include a map in the e-mail. Type the address you want to map, then select it and issue “map” in Ubiquity. In the preview, you'll see a thumbnail-size map of the area (from Google Maps). Refer to Figure 5.

If you execute the command, you'll be taken to the Google Maps page but since what you want to do is insert the map in the mail, click on the image in the preview to get a larger, interactive version. After scrolling and zooming in on this map to your satisfaction, click the “insert map in page” link and Ubiquity will insert the map into your e-mail after the address. Pretty cool, right? Imagine how long it would have taken if you were to do the same without Ubiquity.

Or figure this: What about the times you've been reading an article on the Web and you came across a word you didn't know the meaning of. Select that word, press *Alt+Space* to launch Ubiquity and type ‘define’. It'll immediately give you the definitions of that word, right inside the Ubiquity window. Can it get any easier?

So, basically, Ubiquity gives you a set of commands that make common Web tasks faster and easier. But, as with GNOME Do's plug-ins, the commands that come with Ubiquity are just the beginning—anyone can create new commands and share them. wiki.mozilla.org/Labs/Ubiquity/Commands_In_The_Wild, is a page dedicated to such community generated commands. Think of something, and you will probably find a command for it already there on that page.

You can look at all the commands installed and what each one does by issuing the *command-list* command in Ubiquity. Alternatively, type *about:ubiquity* in the address bar of your Firefox and click on the ‘Your commands’ title (Figure 6). You can also watch the introduction movie on the page to quickly equip yourself and speed up your Web tasks with all the amazing Ubiquity features.

Creating your own commands and sharing it with others is also very easy—simply go through the Developer Tutorial at wiki.mozilla.org/Labs/Ubiquity/Ubiquity_0.1_Author_Tutorial and you should be good to go. Once created, these commands can be embedded in any Web page. If you have Ubiquity installed and you visit a page with an embedded command, Firefox will present you with the option of subscribing to the command (Figure 7). Once you click the ‘Subscribe...’ button, you will be provided with a scary looking warning page as shown in Figure 8. This is very much needed, as an Ubiquity command has full access to your Web browser and can pretty much do anything. While subscribing to a command, you get a check-box saying “Auto-update this feed”.



Figure 8: Warning message stating commands are from untrusted source

This means the commands get automatically updated when a new version is out. Be careful while checking that option, as this introduces a major security risk; just because you decided a command was safe at one point in time doesn't mean that the command will always remain safe! To overcome this, the Mozilla guys are working on creating something called a “trust network”, but this is still some way off in the future. So until then do not install Ubiquity commands unless you are confident that the source is trustworthy.

That said, here's a list of pre-installed commands that you will be using most often in your day-to-day life:

1. *define*: Gives the meaning of a word.
2. *google, flickr, youtube, wikipedia*: Searches your words using the specified engine.
3. *email*: Begins composing an e-mail addressed to a person from your contacts list.
4. *highlight*: Highlights your current selection.
5. *map*: Turns an address or location name into a Google Map.
6. *twitter*: Sets your Twitter status to a message of at most 160 characters.
7. *tinyurl*: Replaces the selected URL with a TinyUrl [tinyurl.com].

Some extra commands that I like are *friendfeed*, *xkcd*, *lolcats* and *websource*. You can install them from the ‘Commands in the Wild’ page.

i Do!

Both GNOME Do and Ubiquity are two wonderful applications, but currently, as the boundary between the Web and the desktop is getting blurred, it would probably make more sense to have one application. I would seriously love to see the clipboard and natural language support in GNOME Do. Then imagine doing things like selecting a portion of text in an OpenOffice.org document and issuing commands similar to Ubiquity like ‘e-mail this to’.

But those days are still a long way off. For now, we have two great individual applications at our service. As someone said, with GNOME Do and Ubiquity, a computer becomes a helper that comes when you call, gives you exactly what you want, and then disappears. I only wish more things in life were that way! 

By Puthali H.B. The author is a programmer at Novell, who loves music and open source. To know more about what she finds interesting these days, go to <http://puthali.googlepages.com>



I'm Running Free... with the Openmoko Neo FreeRunner

The Openmoko project is one of the most interesting efforts with Linux, daringly seeking to free the mobile phone from the grip of proprietary and closed-source software, and to bring the bazaar model of development to this rapidly burgeoning arena.

Openmoko is a Linux distribution designed for open mobile computing platforms (not limited to cell phones). It is also the company behind the Openmoko Linux distribution and the manufacturers of mobile computing platforms, such as the Neo phones. People tend to associate the name 'Openmoko' with the phone (the hardware) itself, but the phones actually have names of their own. The current phone model on sale is the Neo FreeRunner, whose internal codename is GTA02 (the earlier Neo 1973 model evolved up to the GTA01Bv4).

A few hardware changes or patches are still being applied here and there, but with

the FreeRunner, the focus of development has shifted from stabilising the hardware platform to developing the software platforms and user interfaces necessary to bring the phone to mass-market usability levels.

In true FOSS style, there are already quite a few software distributions available for the phone, most of them under intense development. We'll take a look at those soon.

So what's so great about an open phone?

Unlike a 'closed' phone, where the handset manufacturer and network operator combine to prevent you from exercising the full capabilities of your phone (and even from installing software that is

not controlled by the network operators) in order to protect their business and revenue models, you are free to do pretty much what you want on a FreeRunner (excepting restrictions imposed by authorities on phone radios). You have a choice of the software you wish to run, and can even hack together your own, or modify existing software to suit your personal wants or needs.

Exactly how open is the FreeRunner?

All software that runs on the main CPU and that can be updated by the user has its source code available to you under a FOSS licence.

For compliance with FCC rules, though, radio chip firmware are in 'black box' hardware modules that cannot be modified by users (and, of course, is not available in source form). Effectively, these firmware modules are 'hardware only'; the drivers for the hardware, however, are open-sourced. The Openmoko wiki has a page [wiki.openmoko.org/wiki/GTA02_Openness] that explains, component-wise, the documentation available. One notable hurdle is the SMedia 3362 graphics accelerator, for which documentation is only available under NDA.

FreeRunner hardware specs

The FreeRunner's hardware specs are impressive for a mobile phone:

- A high-resolution touch-screen (1.7 x 2.27 inches, or 43 x 58 mm), over a bright and vivid 480 x 640 pixels display
- A 400 MHz Samsung SC32442 System-on-Chip with an ARM920T core
- An SMedia 3362 2D/3D graphics accelerator
- 128 MB of SDRAM memory and 256 MB of integrated flash memory, expandable with a microSD or microSDHC card (the single internal slot supports up to 8 GB SDHC (Secure Digital High Capacity) cards, but is not switchable on-the-fly—it
- requires you to shut down and remove the battery and lift up the SIM gate to get at the microSD gate).
- An internal GPS module, Bluetooth 2.0 + EDR and (via an Atheros chipset AR6001 flash version) 802.11 b/g Wi-Fi connectivity.
- Tri-band GSM and GPRS—Class12/CS4/B 2.5G (not EDGE)—available in two versions: 850/1800/1900 MHz for North America, and 900/1800/1900 MHz for the rest of the world. As of the current model, the FreeRunner is a GSM-only phone; later models may support CDMA.
- Two 3D accelerometers enable automatic re-orientation of the display between portrait and landscape mode, as well as enabling use of 'gestures' to invoke common actions on the phone—such as performing Bluetooth pairing by shaking two phones that are held together.
- Two LEDs illuminate the two buttons on the rim of the case—a blue/orange one behind the power button, and a red one behind the AUX button.
- A 1200 mAh smart Li-ion battery powers all that hardware; despite packing more power than



Figure 1: The shipping package contains...

batteries of comparable size, it is still a bit overwhelmed at present, especially if you have any of the radio components turned on when in a low-or-no-signal area. Better power management code is expected to increase battery life up to 150-200 hours on standby, or a talk time (with backlight off) of 3-4 hours.

Is it fully functional?

The obvious question from someone impressed by those specs is: "How usable/functional is it?" The short answer is best taken directly from the wiki [wiki.openmoko.org/wiki/Neo_FreeRunner#How_usable_is_it.3F]: "As the hacker's dream toy: it is fully functional. As a GSM phone: some people have been using it to receive and place phone calls and SMS for months, but with currently shipping software the battery life is only one day. As a GPS device: critical bugs have been ironed out and there is nice software to know where you are using OpenStreetMap. As an alarm clock, media player, Internet browser, game console, e-mail reader and contacts manager: software is not stable yet. If you want a fully functional smart phone, then download the Qtopia distribution. If you want to help Openmoko develop its end user

applications, then download 2008.8. It's still not feature complete, but it will give you an idea of the direction we are headed."

User experiences

What's in the box?

I mentally praised the packaging standards—the phone was so well-packaged that the parcel could probably have been drop-kicked without damaging the phone... that's a rarity these days!

First out of the package was a bag containing a soft pouch (prominently branded with the Openmoko logo) with drawstring, and a Ziploc bag with the hands-free headset in it. A very nice touch, in my opinion, was a tiny bag containing two sets of spare covers for the earpieces; these are next to impossible to buy on the open market, and I appreciate the thoughtfulness in including these!

Next, of course, was the shrink-wrapped Openmoko box itself. In it, ensconced snugly inside a cut-out within a layer of rubberised Styrofoam, was the phone. There was also the USB cable for connection to a PC; a 512 MB SanDisk microSD card with the adapter for use in SD card readers; the charger, and two different snap-on plugs -- one with rounded pins, one with flat prongs. There was also a nice, solid pen-cum-laser pointer-cum-mini-torch (the white LED is pretty bright in the dark) with a fluorescent yellow little plastic nub—the stylus for the touch-screen. The pen is heavy compared to, say, a stylus for a Palm PDA, but crams in more features—now that appeals to the gadget freak, doesn't it?

A green card that lies atop the phone has a quote from Lao Tze, and the URL to the 'Getting Started' page on the Openmoko wiki. Having the wiki as the documentation means that the documentation can continue to evolve to support newer generations of the phone, and that the company

doesn't spend on printing a manual to put in the box—nice!

Switching it on

My FreeRunner booted the stock 2007.2 distribution when I plugged in the wall charger (see Figure 2). That distribution has been rendered obsolete some time ago, the new 2008.9 distribution being the one to go with currently. (New phones still seem to be shipping with 2007.2, according to the Openmoko wiki—I wonder when they will shift to flashing a newer distribution at the factory.)

A little hiccup is that the phone doesn't charge the battery unless it has fully booted; the charging circuitry is software-activated by code that runs only when the phone has booted. In addition, not even the wall charger's 1000mA is enough power to boot the phone; it needs additional power from the battery to boot. This means that if you've let your battery drain totally, you can't boot the phone, and can't charge the battery!

There are workarounds listed on the wiki, the easiest of which is to borrow a charged Nokia BL-5 series battery from a Nokia phone to assist in booting the phone; once the phone has fully booted, you can hot-swap the flat battery back in, and let it charge. The support mailing list and the wiki also mention people being able to plug in the wall charger to a phone with a flat battery, leave it for half an hour, and then boot. This would imply that the battery is getting some charge—which means that perhaps newer versions of the phone have a fix of some sort. I have, however, seen no official confirmation of this.

The boot process is a long-drawn out affair, taking a few minutes to get the phone to the point where you can use it. The boot time seems to have increased for 2008.8, probably because there's a lot more stuff in it than was in 2007.2. I guess the idea is to not shut the phone down unless you're running



Figure 2: FreeRunner booting the stock 2007.2 distribution

really low on battery. As of writing this, however, suspend/resume do not work too well, with some users reporting failure to suspend, failure to resume, or sub-systems like GPS or GSM not working after a resume. That is certainly going to improve, but for the time being, power management is an area that still needs a lot of work. At present, the battery barely lasts eight hours; even less if you make calls or have GPS, Wi-Fi and Bluetooth turned on, or are in a low-signal area.

There is still quite a way to go before the phone becomes an end-user/mass-market phone. People report issues [wiki.openmoko.org/wiki/FAQ#Neo_FreeRunner_Known_Issues] with several core functionalities including call quality, and the ability to make calls or send and receive SMS messages; GPS reads fail often; you need to be outdoors to get your first fix, and that takes some minutes during which you can't move the phone; Wi-Fi connections have hiccups; and every now and then you get a broken package or two in the package feeds that trigger a spurt of mails to the support mailing list. The phone is right now meant only for developers, but what a huge window of opportunity is available to jump in and help develop the software!

Administering the phone

By default, you're expected to connect the phone to a host desktop/laptop via USB. As usual, instructions for configuring networking on the host computer are on the wiki. The phone uses *dropbear* to provide a tiny SSH server, so you can SSH into the phone to administer it. With the 2007.2 distribution, this was practically the only way to update the package manager's package listings, upgrade the software on the phone, and install new packages.

The 2008.8 distribution provides a graphical package manager on the phone, but the packages listed are a small subset of those actually available via the feeds.

Here are a couple of 'features' I found a little irritating in the 2007.2 package manager: you can run the package manager, *opkg*, in test mode; it downloads packages in the test mode too, although it doesn't install them. However, instead of asking the user whether to keep them around (for a subsequent upgrade or install), it happily deletes the package files (megabytes of them, sometimes).

To actually upgrade or install, you have to wait while packages are re-downloaded... there are at least a couple of possible fixes for this. First, create a temporary package cache on the (the default is 512 MB) SD card. The second would be to create a package-caching utility on the host computer; since desktop distros like Debian/Ubuntu already have apt-cache, a modified version of that utility (to handle the *ipkg* files for the phone) should be possible. A second minor irritation would also be solved by either of these measures: for packages that were already on the phone—the identical version—the package manager seemed to download the packages first, then decide that the exact version was already installed, and discard the package!

In my case, after the first-time upgrade that the wiki advised, the package manager tried to restart the X server (main GUI), but it failed to do so for several minutes. Tired of waiting, I rebooted the phone, which was then unable to start the X server at all! This was my first experience with package management boo-boos, and it wasn't very pleasant. It underscores the fact that right now, and for the next several months, the phone is only for power-users and hackers, and not end-users.

Anyway, since at the time I was a gross n00b at this, I headed for the #openmoko channel on the Freenode IRC server. There, one of the gurus suggested that I uninstall the *gtk+-fastscaling* package, a conflict with which was preventing installation of a *gtk+* package that newer GUI packages required in order to work. I needed to force the uninstall, because *opkg* warned me that 'core' phone packages depended on it... then installed the *gtk+* package, and after a reboot, had my GUI back. This turned out to be just the first of quite a few little irritations and alarms—but then, this situation is bound to change within a year.

Make Money With Free Software

- ▶ Its Recession.....Every company is cutting cost.
- ▶ FOSS saves 80% plus costs for users.
- ▶ Free Software Movement has now created full arsenal of award winning products to fulfill variety of business needs from across the sectors.
- ▶ There is a huge requirement of services for implementation and support of such products.
- ▶ Infact many companies are now searching vendors for FOSS services.
- ▶ Its your opportunity to start a great business with minimum investments.
- ▶ (i.e - One PC/Laptop, 7 days of training, Some cash for travelling. Freshers will need more time.)
- ▶ If you are a firm dealing in hardware, support, website development, in sales with business contacts this can prove to be a perfect match.
- ▶ About us - Located in Pune. With more than 100 peers working on different FOSS products, IT Strategy and Process Redesign.



Join us as

1. Business Partner - sales, implement solutions
2. Core Team member - design, implement, support.

To Join Call/SMS now : Paritosh +91 9922904551
Email : paritosh@wikocean.net URL: www.wikocean.net

www.eleb2b.com

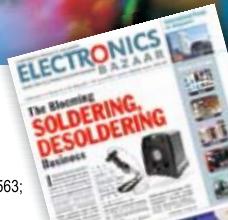
Electronic Components Products Machinery Will now be sold here

Presenting Electronics Bazaar,
India's First B2B Sourcing Guide
For Electronics Components,
Products and Machinery.

Unit Price: **Rs 50**

Subscription Rate: **Rs 350 (12 Issues)**

EFY Enterprises Pvt Ltd, D-87/1, Okhla Industrial Area,
Phase-I, New Delhi 110 020; Ph: 011-26810601-03; Fax: 26817563;
Email: info@efyindia.com; Website: eleb2b.com



Distributions available for the FreeRunner

There are several distributions for the FreeRunner, enough to make it a bit of a chore to try out all of them; a list is available on the Openmoko wiki [openmoko.org/wiki/Distributions]. Most new users find FDOM (FAT and Dirty Openmoko) the most comprehensive and functional distribution, while Qt Extended/Qtopia offers a functional smartphone. The recommended platform for developers is FSO (FreeSmartphone.Org), which offers a choice between EFL, GTK+ and Qt GUI toolkits, and Java, Python or whatever open programming language you like. The SHR (Stable Hybrid Release), Debian, Gentoo and Android distributions are for

people who do not need to ask which distribution they should use. In other words, you're on the cutting-edge if you try these distros, so be prepared to shed some sweat and tears, if not blood. The support and developer mailing lists frequently provide help on distribution-related questions, however.

Official distribution and community-driven variants

The 'official' distribution currently out from Openmoko Inc is the Om 2008.9 Update, which is a minor upgrade of Om 2008.8 (formerly named ASU). Users with basic telephony needs should find it tolerable to use as an everyday phone.

FDOM is a community distribution that adds many fixes and applications

to the Om 2008.9 distribution, while retaining the ability to update through the official feeds.

SHR contains some basic GTK+ based applications which make use of the FSO, and elementary EFL (Enlightenment Foundation Library) dialer, messages and contacts applications, programmed in C. As of writing, there is no stable release of this distribution yet.

Third-party distributions, including Android

Qt Extended, formerly known as Qtopia up to version 4.3.x, comes from Trolltech, which is a Nokia company, and the home of the Qt cross-platform application framework. It aims to provide a ready-to use image for Openmoko

Getting the FreeRunner to India

As mentioned on the Openmoko commercial website [www.openmoko.com/distributors-asia-india.html], IDASystems [idasystems.net] are the Indian importers and distributors of the Openmoko. I was curious to know why India was favoured as one of the first countries in the world to have the FreeRunner available for purchase (far before the USA, a market that most manufacturers head for first!). Besides that, I also had other questions, which I put to Rakshit Hooja of IDASystems, and he obligingly answered them.

Some information on IDA Systems

IDA Systems was founded in 2000, mid-dotcom-boom, ran a small software development wing, and also sold handheld computers (including Psion and Casio). Business was not that great; IDA was too early with the handheld computer idea in India. The dot-com bust brought the company to a virtual standstill. In 2003-4, Hooja acquired a stake in IDA, and became a director of the company, along with Amiya Kumar Das.

Hooja holds a master's in Sociology, and an M.Phil, from the Centre for Studies in Science Policy, JNU, on open source software and the needs of developing countries. He first used Linux in 1994/95, and has been dabbling in technology-related things since then. When at the M.Phil stage, Hooja volunteered to help spread Firefox (or rather, Phoenix/Firebird as the browser was named then), and realised that he wanted to do something with free/libre and open source software (FLOSS). Doing the M.Phil, however, made him realise that academics was the wrong arena in which to work on FLOSS; he should actually do something in mainstream society. At IDA, he put forward the lofty aim of assembling a Linux phone in India, and started raising capital. The fruits of this effort are what is now the IDA 2B1 [idasystems.net/ida_2b1], a prototype of which is expected to be available in December 2008.

Q While India is certainly a huge market, what were the other factors that led to the choice of India as one of the first countries where the FreeRunner went on sale, even before the USA?

While doing research for the IDA 2B1, I came across Openmoko, which was at a much more advanced stage and had funding. I contacted Sean [Moss-Pultz, who is CEO of Openmoko Inc now] and the rest just followed.

Why is it in India this early? I guess because I believed in the idea (not just the FLOSS ideals, but that it could be commercially successful as well). Also, I felt that such a device would be—no, is!—ideal for the large IT pool

in India to work on and customise to create new solutions, bearing in mind that what a developing country needs from a mobile/portable computer may not be the same as what Nokia or Apple, for instance, are designing. Because of this, I convinced a few private investors, and put up advance money with Openmoko (no one else in Asia had been willing to take that risk; we had to wait a few months before the first deliveries, too).

Q How many units have been sold till date? How many of those seem to you to have triggered contributions to the Openmoko effort?

We've sold between 50 and 100. We're pretty open with sales figures, but I don't keep a daily figure, as it kind of makes one lose focus of the one-year target of 2,000 units. About 75 per cent of buyers seem to be lurkers on at least one of the Openmoko lists.

Q I strongly feel that the FR (FreeRunner) is definitely not for the technologically-challenged; it's a geek's phone. What's your take on this? Do you have any estimates on when the software would get end-user-friendly enough to actually become a mass-market phone?

It is not a mass-market phone at all, nor is it being advertised as such. It is a Linux computer with (currently) flaky GSM, GPS, Wi-Fi, GPRS and Bluetooth support. I have been using the FR from the early Neo 1973 days, and have seen the flakiness decreasing constantly. So I would guess we will have stable software by the end of this year, but a GUI that your grandma can use and be impressed by—not before February/March next year.

Q What warranty are you currently offering on the FR? In case of known hardware problems or new workarounds, will you accept earlier-built phones returned to you for upgradation or implementation of hardware fixes? Also, what other hardware support is in place if, for example, one damages the screen? Do you have hardware facilities in India, or would the phone be

devices, and features a noticeably robust telephony stack. The recent 4.4.2 release also sports a WebKit-based browser and Gtalk support.

Debian, 'the universal operating system', has thousands of packages (though most of them are designed for desktops or servers). As Joachim Breitner of the *pkg-fso* team puts it, Debian for Openmoko is not really a distribution, per se, but rather a different underlying system for Openmoko distributions. It ships software from the FSO stack, but provides more packages (and takes more space!).

Gentoo is a fast, modern meta-distribution with a clean and flexible design. Gentoo's packaging system uses source code (although support for pre-

compiled packages is included too), and it lets you choose how much you want to compile yourself, how to install it, and much more.

Android, Google's famous mobile phone platform, has Openmoko's full support in getting it to run on the FreeRunner. Currently, porting efforts are under way, and more information is available on the wiki [wiki.openmoko.org/wiki/Android and wiki.openmoko.org/wiki/User:Seanmcneil3]. The first page has links to a couple of YouTube videos showing Android running on the FreeRunner.

Trying out new distributions

So how does one go about trying out all those distributions? Well, to answer

that, let's start with a little background information.

There are three main downloadable and flashable software components involved in the FreeRunner boot process: *u-boot*, kernel, and *rootfs*. *u-boot* performs a role similar to that of the GRUB bootloader on a PC—it loads a kernel image into memory, and passes boot parameters to the kernel (including the device on which the root filesystem is located). As the kernel boots, it initialises the hardware, and then mounts the root filesystem. The kernel then runs */sbin/init*, which handles the rest of the boot-up sequence, such as displaying the splash screen and progress bar. This sequence is the same whether the device is booting from built-in flash memory or

shipped back to FIC for work?

Openmoko only offers us a 28-day 'Dead on Arrival' RMA option. We are currently offering customers a 30-day 'Dead on Arrival' (DoA) warranty, but are in the process of finalising our warranty terms for a one-year repair warranty arrangement. We have replaced the phone where a genuine hardware defect was found (for example, when the SD card socket did not work).

We have arrangements with two hardware repair facilities in Jaipur and Delhi for the purpose of repairing the FreeRunners if required, but DoAs we send back to FIC.

Q Do you plan to have any buy-back offers for early adopters who wish to upgrade to newer releases of the phone? If so, what might these offers be like?

We will be offering buy backs—about 50 per cent of the price—when full model numbers are upgraded (for example, if you have a GTA02, then we will offer a buyback when the GTA03 is put on sale). We will then refurbish the older phones and sell them again in different markets, at lower prices. Refurbishment includes minor factory changes, which may offer significant user benefits without raising the base price substantially.

Q What are your plans for supplying value-addition bundles to FR customers? For example: SDHC cards, rollable/foldable keyboards with a mini-USB connector, spare battery, external charger, screen protector films, external GPS antenna, etc?

Yes, we plan to do this, but only when mass-market sales start. We won't sell the keyboards, though, as they are easily available from third-party sources. We will just list the tested ones on our website.

Q Do you, personally, use an FR as your main phone? And do you try out new images and options for the phone?

Yes, I use it as my main phone—but then, I used the Neo 1973 [the previous hardware version] with

all its problems, as my main phone too! I keep trying the images out, but have not tried Debian yet. As the main phone I use Qtopia (installed into flash memory) with my music on the SD card. I'm lucky to be the person with the largest supply of FreeRunners in India [laughs] so I carry another FreeRunner with me, which runs FSO 2 and Tango GPS, with the maps on the SD card.

Q Do you or IDA as a company hack any of the packages or contribute any software to the repositories?

I personally don't, but our developers are working on *dfu-utils* [used to flash new distributions to the phone] for Windows, along with a sync manager for Windows (unfortunate, but a highly-requested feature for India).

Q What about any concerted effort to localise/adapt the software to make it a highly portable computing device for rural India? Has any agency taken on the challenge to spearhead this effort?

There are plans to localise the FR; with full Debian installed, Indian fonts do run on it. For rural deployment, we have an arrangement with a leading Indian cellular provider (the name is confidential) to provide low-cost GPRS SIMs to us in Rajasthan, and we are going to be testing them over the next few months to see how the FR functions as a highly-portable computing/communications device in rural areas.

The purchase and delivery experience

IDA Systems made it easy to purchase the phone, since a credit card wasn't required. I could simply deposit a cheque in the company's HDFC bank account, after which, I mailed Hooja and Zoheb Ansari (the latter handles sales enquiries—zoheb@idasystems.net) the cheque details. The phone was dispatched the next day, after verifying that the payment had cleared. After a reminder e-mail, Zoheb mailed me the consignment information. Except for a mysterious delay of two days in Blue Dart's local delivery, I got it pretty quickly. The purchase process was quite painless, all in all.

from the SD card. The differences are only in how the kernel is loaded (from NAND flash or from SD card), and which device is mounted as the root filesystem.

The FreeRunner has a separate 16 MB NOR flash memory that stores a 'fail-safe' copy of the *u-boot* bootloader. In the event that you happen to corrupt or 'brick' the main NAND memory bootloader, you can only overwrite the fail-safe image using the (optional extra purchase) debug board, and not via the USB cable (standard method of connecting to a host PC). Thus, it is possible to write not just new kernel images and root filesystem images, but also a new bootloader version, to the NAND flash memory, without fear of bricking your phone. The earlier version, the Neo 1973, lacked this fail-safe feature, which meant that flashing involved some risk of bricking the phone. The FreeRunner is worry-free while trying out new distributions.

Additionally, instead of flashing a new distribution to the phone's main NAND flash memory, you can install it on the microSD card and choose a *u-boot* option to boot from the SD card. This makes it easy to keep the distribution that you'd like to use most of the time in flash, but to still try out other distributions just to see what's new with each. The procedure to put the distribution on the SD card is different from flashing the phone's memory, though. Note that at least earlier versions of Qtopia assumed that the distribution would be installed to flash memory; if you put it on the SD card, you could expect hiccups like your media files not being found and so on. The version of Qtopia that I had tried out also lacked a folder browser dialogue box, so that you could navigate the filesystem to locate your files. You are required to manually edit configuration files. Hopefully, newer versions will have done away with these small issues.

To try out a new distribution, you would generally:

1. Download the kernel and rootfs (root filesystem) images for

the distribution. (Openmoko recommends that you do not update *u-boot* unless they release a major new version with bugfixes.) The 'Distributions' wiki page [wiki.openmoko.org/wiki/Distributions#Images] has links to separate pages for each distribution; each of those pages has download links to the images for that distribution.

2. If you are going to install the distro in the main flash memory and don't have the DFU (Device Firmware Upgrade) utility, download it. Additionally, you can download a graphical interface (GUI) for the DFU utility, if typing commands at the command prompt irks you. Both of these, and lots more information, can be obtained via the page wiki.openmoko.org/wiki/Flashing_the_Neo_FreeRunner
3. Based on whether you're installing the new distro to NAND flash or to the SD card, follow the appropriate procedure at wiki.openmoko.org/wiki/Flashing_the_Neo_FreeRunner or wiki.openmoko.org/wiki/Booting_from_SD to get the new distribution onto the phone.
4. Reboot the phone, and choose the appropriate boot option (the default 'Boot' option if you put the new distribution into NAND flash, or the 'Boot from SD card' option if you put it on the card).
5. Try out the new distribution, of course. This involves installing your favourite applications on the new distribution, naturally, and (if you had to re-partition and format the SD card) copying your media files, etc, back to the card.

A quick peek at 2008.8

As we've seen, there are several distributions, and I couldn't possibly include a peek at images from all of them—so I'm just putting in a few image captures from 2008.8 (I haven't yet upgraded to 2008.9) to give you an idea of what the user interface is like.

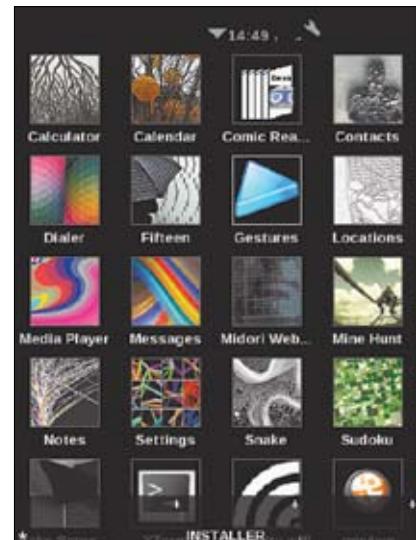


Figure 3: Home screen with a list of icons of the installed programs



Figure 4: Top shelf pulled down

After installing a few programs onto the phone, the list of icons (shown in Figure 3) naturally overflows the screen. You can tap in the blank area between the icons and drag upward to scroll to the lower, unseen icons. No scrollbar shows up, though. The plus signs in the translucent bar at the bottom are placeholders for buttons that will come up soon. The 'Installer' text in the same area launches the package manager when you tap it.

The pull-down shelf at the top (shown in Figure 4) offers clickable gadgets to launch utilities such as the settings manager (the tiny spanner icon); it has gadgets to show time,



Figure 5: Contacts application—a list of contacts

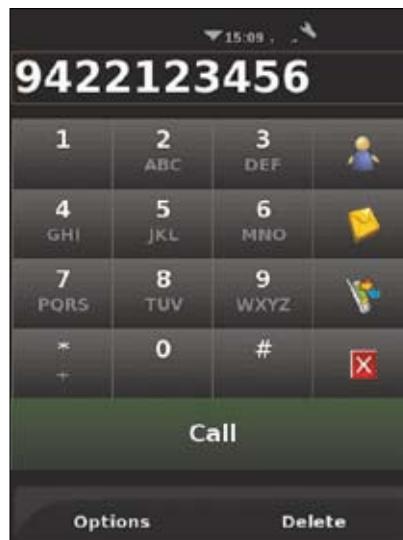


Figure 7: Dialler program



Figure 9: Messages application



Figure 6: Contact details screen

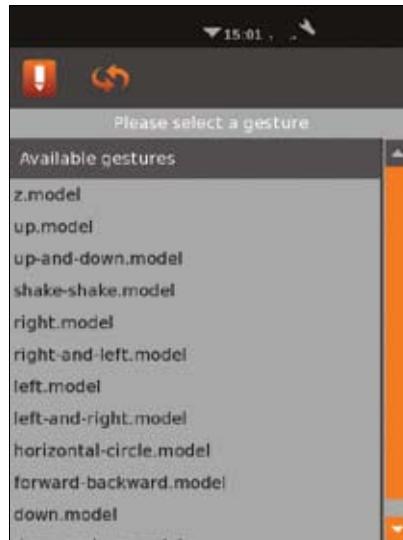


Figure 8: Gesture training application



Figure 10: Day view in the calendar application

and battery and wireless connectivity status (GPS, GPRS, Wi-Fi and Bluetooth were turned off). The three lines in large text are a list of running programs, each of which you can click to switch to. The tiny 'REMOVE' item at the lower left of the shelf closes the currently active program (except for the *Home* screen, which must always be running).

I have a now-unused Hutch SIM card in the phone currently, so it shows only a few operator-related contacts (see Figure 5). The 'Options' menu has items that let you create a new contact, send all contacts to another phone, manage contact groups, and choose the storage area

from which contacts are shown—SIM service numbers, contacts defined on the phone, and contact numbers from the SIM card.

Tapping a contact brings up the contact details. The first, the *Overview* tab, is seen in Figure 6. The other three tabs are *Details* (displays the phone number) for the contact; call history for this contact; and a messages history. The *Options* menu is identical on all four tabs. The items let you delete the selected contact, import the contact to the phone, or message the contact vCard to another phone.

If you're thinking that these look a lot like Qtopia, you're right!

The Dialler application (Figure 7) is quite straightforward. The column of icons on the right let you add this number to an existing contact (to save as a new one, you use the single item on the Options menu below), send a text message, view call history, if any, and exit the dialler app. Tapping out numbers is a bit sluggish, though—this is another of those niggling little issues that needs to be ironed out before the phone can go mainstream.

The gesture-training application (Figure 8) lets you 'train' the phone to recognise the way you perform certain gestures. The 'Gestures' wiki page [wiki.openmoko.org/wiki/



Figure 11: Settings—screen 1



Figure 12: Settings—screen 2



Figure 13: Package manager—installing a package

Gestures] describes how to get the phone to recognise the gestures you have 'trained' each model to recognise, and convert them to useful actions. It also includes information on the use of the training application, and links to a video showing gestures being recognised.

The messages application is quite appealing, as you can see in Figure 9. It sometimes doesn't work, but again, that is a bug that will be fixed.

Figure 10 is a glimpse of the day view of the calendar application.

Figures 11 and 12 show options available via the settings application. The names are mostly self-explanatory after a little experience with 2008.8.

Figure 13 shows a package being installed using the built-in package manager. The blocky green effect is due to the fact that it is a progress bar that oscillates from left to right and back, and is difficult to capture. The package details screen is visually quite appealing!

From the images, it should be obvious that the Openmoko team and the community are striving to create a gorgeous interface atop powerful and useful software (far beyond the regular run-of-the-mill phone applications—take a look at [projects.openmoko.org!](http://projects.openmoko.org/)) and there are noticeable improvements with each new version of each distribution.

It is an uphill task to keep abreast of all the distributions, so most people usually choose one that is suited to their current goal for the phone (development platform, usable phone, geek powertool...) and stick with it, occasionally trying out other distributions on the SD card to see what's new with them.

Get involved!

As is the case with open source, there is immense potential for new contributors to jump in and make a difference here [wiki.openmoko.org/wiki/FAQ#How_do_I_join_the_Openmoko_project.3F]. System or userspace code, artwork, sounds, games, themes, customisation, packaging... and much more—this software ecosystem is fertile ground for contributions. It would make an ideal platform, in my opinion, for students and budding software developers to cut their teeth on open source, and would also provide experienced developers with one of the most appealing gizmos—an utterly personalised (or personalisable) smartphone.

You don't even need to buy a FreeRunner to pitch in and contribute; it's possible to do all the development on a PC, and run a QEMU virtual machine of the phone right on that PC to try out your program [wiki.openmoko.org].

org/wiki/FAQ#Development]. The virtual machine isn't a complete representation of the hardware, but is usable for most of the testing needs you may face.

For a look at some of the projects under development, visit *projects.openmoko.org*—the burgeoning number of interesting projects there may well generate ideas for your own killer app!

If you're interested in exploring the FreeRunner as a platform at your college or university, do go ahead and mail Rakshit Hooja (rakshit@idasystems.net) with your enquiries; he is enthusiastic about spreading open computing platforms in India, and may be able to work a special purchase deal for your educational institution. 

By: Edgar D'Souza is a FOSS fan, technical writer and editor, and has also done systems administration and software development for a living at different points in his life. Please send bouquets or brickbats (small ones, please!) to edgar.b.dsouza@gmail.com

This article includes content taken from the Openmoko Wiki (wiki.openmoko.org), which is licensed under the GNU Free Documentation License 1.2. Other parts, and images, are (c) Edgar D'Souza 2008. The whole article is released under the GNU Free Documentation License 1.2.

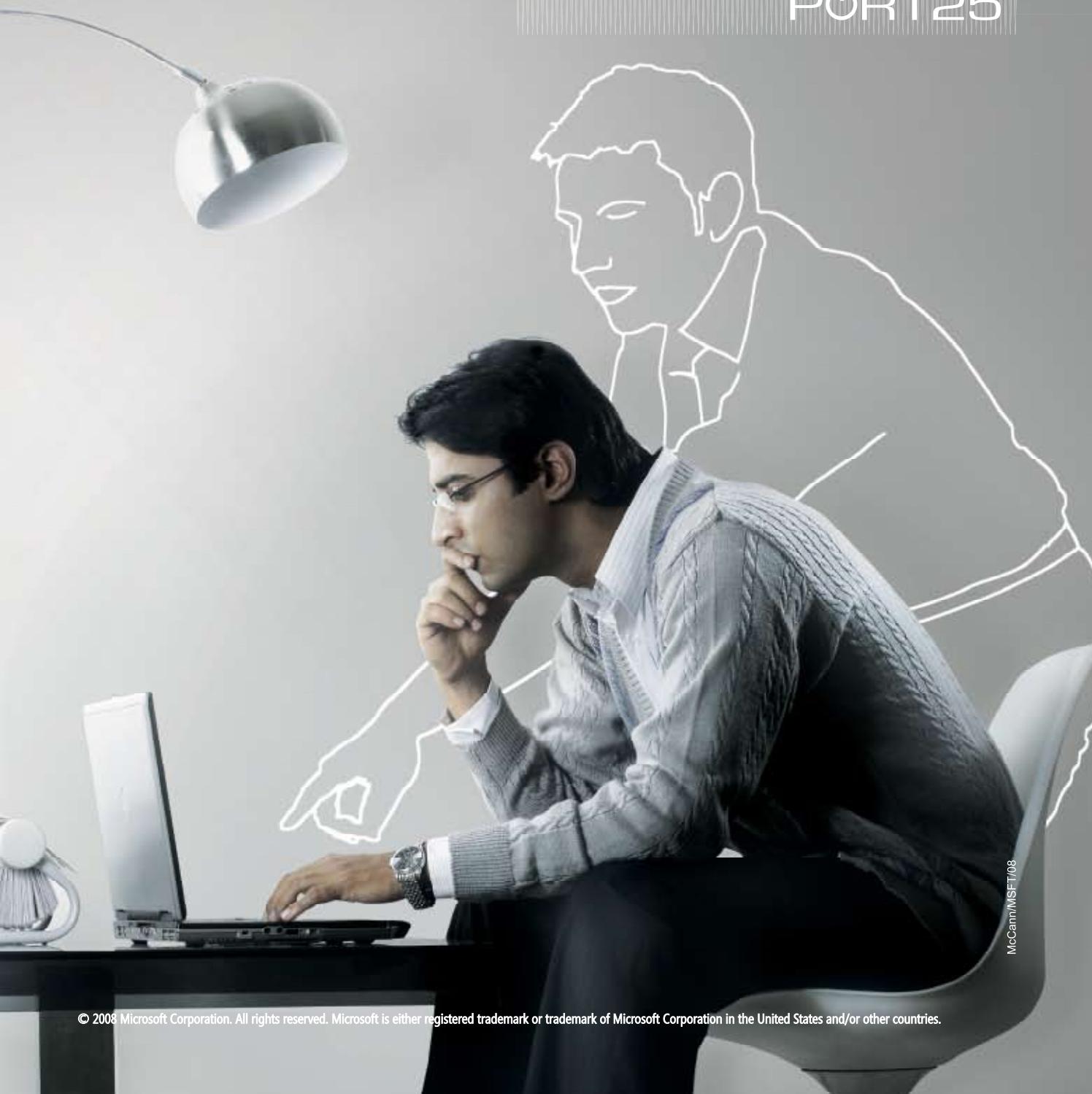
Microsoft®

EQUIP YOURSELF

Microsoft invites all developers to join the open source community and online resource centre. Interact with the likes of Sam Ramji, Director of Open Source Software Lab, Jamie Cannon, Lead of OSS Community & Platforms and many others to find solutions to all interoperability challenges.

JOIN WWW.PORT25.IN, A MICROSOFT PLATFORM FOR OPEN SOURCE DEVELOPERS.

PORT25



McCann/MSFT/08



The Intrepid Ibex Awaits Command

Ubuntu 8.10 is out! We give you a sneak peak...

One thing's for sure—no matter what good or bad is said, written or done about Ubuntu, it still changed the way people looked at Linux for the desktop. I know a set of people who use Ubuntu but have no idea that it is Linux, or for that matter, even care. Taking this mission further, Ubuntu 8.10 'Intrepid Ibex' was released on October 30, 2008.

The big ones

As with every update, each of the core packages has been updated (Ubuntu starts with the Debian unstable tree to 'make' a new version). Some important ones are:

- Linux 2.6.27
- X.org 7.4
- GNOME 2.24
- Network Manager 0.7
- Samba 3.2

...and lots of others.

Of course, there are a plethora of new features, more packages added to the repositories, and the usual patches-here-and-fixes-there. I won't be taking the normal review route, so there's no 'How To Install' here. Instead, I'll focus on what's changed under the hood (by the way, the installation procedure hasn't changed over the last edition) because that's what really matters!

The interface. 'New and improved!'

The new GNOME 2.24 brings in a lot of small but effective improvements. The important ones are those to File Roller (the archiving utility) and Nautilus (the file manager). File Roller now has support for ALZ, RZIP, CAB and 7Z formats—so there's no more separate 7-zip installation!

Nautilus has a lot more to offer. A very cool feature is *tabbed browsing*. I agree this was one thing I had been waiting a long time for, since tabs are now featured everywhere (thanks to Firefox). I can see the KDE people saying, "We've had them for ages!" but *hey*, now so do *we*!

The Trash window has a *Restore* option that is (strangely) something I missed from Windows! The *Places* sidebar has OS X-inspired *eject icons* for

© Nino Barbieri. This image is licensed under Creative Commons Attribution 2.5 License.



Figure 1: The Ubuntu menu and the System Monitor app

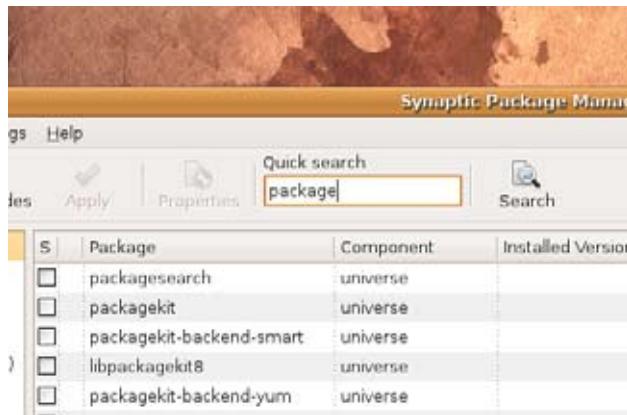


Figure 2: Quick search in Synaptic package manager

removable drives. I found them a bit strange, as they don't have the 'click' visual cue and you get confused about whether it has been clicked or not. But they work!

Also, Nautilus now has an 'Archive Mounter'. What it does is mount any archive as a removable drive. It will display the archive on the desktop and in your *Places* menu. In my opinion, that's not of much use, since File Roller does the same thing with a different interface.

Apart from these, I found other changes as well, which I couldn't figure out from my hour-long trial but are documented on the Internet. This includes a 'Delete permanently' option and a 'Compact view' option among others.

Let's move to the *Panel*. The *Panel* now has a nice, new Fast User Switcher applet that does some very cool things.

1. The *Shutdown*, *Restart*, *Standby* and other options got a nice visual makeover (which I was told were taken from openSUSE). Nice to see the love across distros.
2. When you're using an IM client like Pidgin, it automatically integrates your presence settings like *Available*, *Busy* or *Away* into the applet, adding it to the other options of *Shutdown*, *Restart*, etc. Nice!
3. It includes an option for Guest Sessions, so you know what to do the next time your friend wants to check out your computer. The sessions are secured with AppArmor so your data is safe! Just log out the Guest and you'll have your machine back safely.

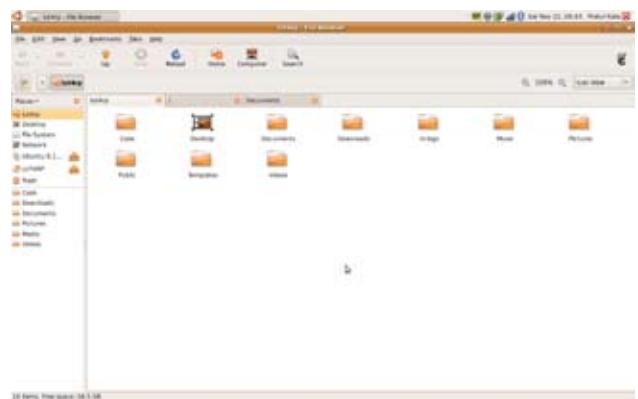


Figure 3: Tabbed interface in Nautilus file manager

More goodies

Let's see what else got a face-lift, or rather 'code-lift'.

Network Manager 0.7 was another anticipated release. This improved tool makes managing multiple networks simultaneously very easy. It allows for wireless, wired, 3G and even PPPoE connections from one single window. It has better support for hidden wireless connections and route management as well. Strangely, I also thought it detected wireless networks quicker.

Samba 3.2 got IPv6 support, which though nascent, might be useful to many a systems administrator. It also includes support for encrypted networks, clustered file systems and Windows Vista. I couldn't test any of this out, but from what I read on the Web, it's impressive.

Synaptic Package Manager now has a quick search. I liked this, since I don't use the main Synaptic search anyway. If you've got many repositories, it tends to freeze for some time. I used to type in the letters directly to reach the specific packages, which the developers seem to have sensed.

Seahorse, or 'Password and Encryption Keys' as we know it, had its interface refreshed. Gone is the six-tab interface, which has been replaced with a simple two-tab one. I am pleased to note the developers 'decrypted' that! :-)

Job well done!

You say, "What else?" Quite a number of people across the community demanded a change from the brown, human theme, so we have a new theme called 'Dark Room', which I found awesome. Also, the new X.org is so cutting-edge that many people found their graphic cards weren't yet supported. Hopefully, that will be fixed ASAP.

Overall, it's as fine a release as ever, and the path to world domination and squashing Bug #1 is getting better and better! 

By: Pratul Kalia. The author is an open source hacker and evangelist. He has been using/tearing up computers since 1996. Currently, he contributes to Drupal, and is a maintainer for Drupal.org and the Ubuntu India forums. He lives on the WWW at <http://pratul.in> and is also known as lut4rp.



Gets a Version Older



This much-awaited release of OpenOffice.org brings tighter integration at the suite level. It also unleashes a streamlined user interface and enhanced support for popular document formats. Does the new version make a viable option for enterprises?

OpenOffice.org has remained (arguably, of course) the most popular piece of open source software for the better part of this decade. So much so that, for a good fraction of the users out there, open source software does not exist beyond OpenOffice.org. Thus, any major release of the suite generates a buzz, euphoria and great expectations. Unfortunately, it also draws wild comparisons with pricey proprietary office productivity suites, and that is where things turn ugly.

Top view

My curiosity swelled as I downloaded OpenOffice.org 3.0 amid news that the servers had experienced seizures a few hours into the release due to overwhelming traffic. That this is the first proper OOo release for Mac OS X probably explains the rush. When I ran the new release for the first time, it seemed to load faster than its ancestors. The OOo user interface has apparently gone under the knife, and is now well integrated at the suite level. The nifty Start Center (Figure 1) not only lets you launch any of the OOo component applications, but also gives you quick access to the ever-growing library of templates and extensions on the Web.

Coming to the headcount, OpenOffice.org 3.0 does not bring any new components to the suite, unlike 2.0 that added OOo Base. The core arsenal remains the same—Writer, Impress, Calc, Draw, Math and Base.

The goodies

As far as Web 2.0-style collective authoring goes, OOo seems to be moving in the right direction. You can now create

spreadsheets in Calc and share them with other users (Figure 2). Any edits that they make can be quickly incorporated into the original file. I am hopeful that this functionality will soon be extended for Writer documents and Impress presentations.

Amid other notable enhancements is support for Microsoft Office 2007 and 2008 binary file formats. I tried importing a *.docx* file in OOo, and the conversion worked beautifully. ODF 1.2, the upcoming version of the OpenDocument Format, is also fully supported. Thus, OOo now supports all of the three major document formats that are ISO standards—ODF, PDF and OOXML.

OOo 3.0 also incorporates a solver component in Calc that optimises scenarios where the value of a cell needs to be determined based upon constraints provided in other cells. The marketing pages of www.openoffice.org make a tongue-in-cheek remark taking on the office suite from Redmond: “The new solver component should be particularly interesting to Mac users considering that Microsoft Office 2008 for Mac OS X apparently does not include a solver feature anymore.” Not related to the Linux version of OOo, but interesting nevertheless! Multiple chart-related enhancements have also been introduced in Calc, including support for custom error bars and regression equations.

The OOo team has revamped the notes feature in Writer to make it more usable and intuitive. Notes now appear on the sides of a document instead of showing as yellow rectangles right within the text (Figure 3). Image cropping in Draw and Impress has been revisited as well. You can now crop images by simply dragging their handles as much inwards as you want; no convoluted procedures there! Working in long



Figure 1: The OOo Start Center

documents in Writer should be a breeze now, thanks to the new zoom slider and the much-anticipated functionality to view multiple pages simultaneously in different layouts.

Besides the cosmetic changes, OOo 3.0 also provides enhanced support for XML and XSLT-based filters. The programmability enhancements in this release mean that developers can write add-on applications (extensions) that run atop OOo. For instance, developers can write an extension to generate online help formats from a bunch of Writer documents. The possibilities are endless.

In addition to the six core components, OOo 3.0 also provides ready-to-use extensions and complementary tools. There's a Wiki Publisher extension that facilitates the creation of wiki pages on MediaWiki servers. There is also a Report Publisher extension that you can use to create reports for Base databases. Extensions and tools are added to the online repository at regular intervals, so there will always be dope for you to download and use.

Dreaming on...

To me, the one application that has been most conspicuous by its absence in OOo is a counterpart to MS OneNote. I've grown fond of OneNote and the way it helps me organise my disarrayed thoughts. FreeNote, anyone? One wild thought that crosses my mind is integrating FreeMind, that wonderful free mind-mapping software, with OOo. I'm going to file it as a feature request!

While the OpenOffice.org website directs the user to download Mozilla Thunderbird and Lightning as the calendar and e-mail client extensions to OOo, these applications haven't been customised for integration with the suite. MS Outlook seems to be the top reason why enterprises are finding it hard to replace MS Office with OOo. Coupled with MS Exchange, I see it deeply entrenched into the workflows of many organisations, often with multiple third-party plug-ins providing additional functionalities. I am sure the OOo community will address this void sooner than later.

Also, the OOo team seems to have touched OOo Base

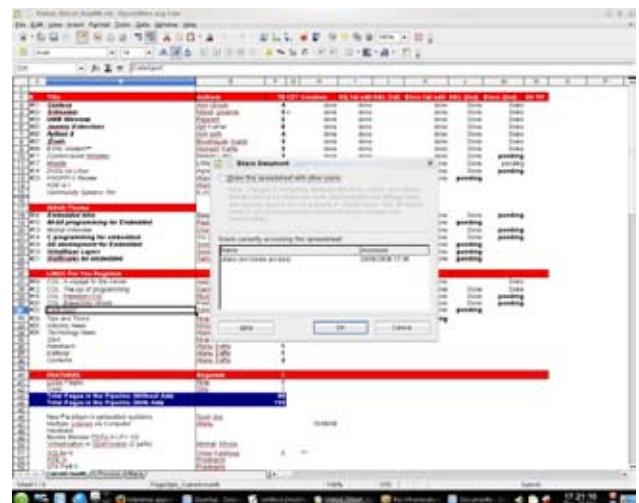


Figure 2: Share spreadsheets in Calc

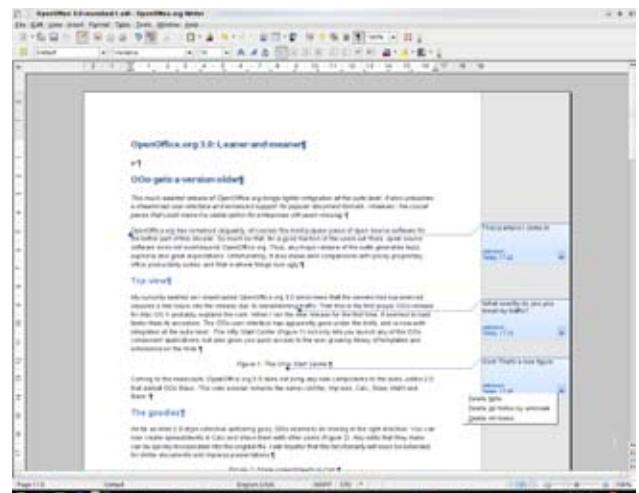


Figure 3: The revamped notes feature

very little since it was first incorporated into OOo 2.0. While the 'queries within queries' feature rolled out in OOo 3.0 is good to have, users have been hoping for greater compatibility between Base and MS Access, as well as improvements to the OOo macros language to facilitate more streamlined access to database features from within the other suite applications.

The verdict

OpenOffice.org 3.0 is a worthy update to office suite. It delivers a bunch of enhancements, but probably not enough crucial breakthroughs for enterprises to consider it ready for prime time yet.

I would rate it 3/5. For the integration, the UI and the freedom! 

By: Samartha Vashishtha. The author is a poet, technical writer and intermittent journalist. He works on the technical communication team of a leading software company. An anthology of his creative outpourings is at <http://www.samartha.tk>. His Hindi blog at <http://samarthav.blogspot.com> draws a significant readership.



Interview

For U & Me

A Walk to Spread the Message of Freedom

Cherry George Mathew (left) and Sooraj K. being led to the IHRD College of Engineering, Attingal. Behind the camera is Anoop John.



*'To claim, to ensure and
to preserve freedom!'*

Walking from the northern end of Kerala to the southern end, covering more than a thousand kilometres, seems like a crazy idea, doesn't it? But for these young IT professionals it was a Freedom Walk—a walk to promote Free Software as an empowering agent for social and environmental activism. On October 2, the Gandhi Jayanthi, these youngsters started walking from Kazargode, the northern district of Kerala. On their way, they visited educational institutes, government offices and met people from all fields of life, and explained the importance of Free Software in all aspects of daily life.

Anoop Johnson, a Free Software activist and CEO of Zyxware Technologies, with his friends reached Trivandrum on November 14th. That makes it a 43-day long Freedom Walk. We talk to Anoop here to get a gist of their experience.

Q. What exactly was the motivation behind the Freedom walk? What were the objectives?

There are two specific objectives to the freedom walk: one is the social aspect, and the other is the technology aspect.

The social aspect is that we believe we can work towards changing our society for the better, and towards solving social and environmental problems by making small incremental changes in our lives, thereby cumulatively making a massive impact in the system. This idea is based on Gandhiji's message: "Be the change you wish to see in the world!"

The technology aspect is that we believe free software is the way to the future because it enables even the common man to access technology, and brings technology as a great leveller for people to access information and services.

Q. Software being a technical subject or something that does not touch many of the common people directly, how difficult was it to explain the objectives? How far did you succeed in putting across your message?

Note that the main objective of the freedom walk was to take the principles behind free software, to the general public. This is, therefore, more general than just free software, and pretty much nothing to do with software as a technical subject.

The principles behind free software are generic enough to explain to the common man. We spoke to people in all sorts of vocations—from auto rickshaw drivers to school teachers. Strangely, people who were less educated could appreciate the right to access of information better.



▲ (R-L) Anoop John, Sooraj K. and Cherry George Mathew being welcomed by students of IHRD College of Engineering, Attingal.

It is hard to assess how far we got through though, since the audience we addressed were a good cross section of the society. We faced reactions from the general public ranging from scorn to enthusiasm. On the whole, however, people did seem to appreciate our efforts.

Q. The team visited educational institutes, government offices, etc. How much awareness about free software and its ideology did you notice at these places?

GNU/Linux is a subject offered by the IT@school program in all government-run schools in the state.

Therefore, school students showed immediate recognition when we referred to this program. All we had to do in such cases was to update them of the history and motivation behind the free software and open source movements, which culminated in their curriculum.



▲ Anoop John addressing students at Government High School Pilicode, Kasargode

The situation in two government-run services we interacted with—the police service, and the KSEB—were, however, in stark contrast to each other. The Kerala police service seemed to be struggling when it comes to support for free software, and therefore its adoption. Free software user groups have realised this shortcoming and are gearing up to provide support to Kerala police. Awareness seemed to be lacking down the ranks, although higher ups seemed to appreciate the potential that free software has for the organisation.

The KSEB (Kerala State Electricity Board), on the other hand, is astonishing in its adoption of free software, as well as its active development of internal projects using it. The 'Oruma' billing software package is entirely developed internally by an in-house team. This is amazing for a non-software company. The story

doesn't end there though. KSEB is actively interacting with the local free software community, to support them technically. We even heard an interesting story of how a team of KSEB developers once gatecrashed into a local FOSS event!

Q. What was the response from social activists about using FOSS as a technology platform for their activities?

Most social activists we spoke with were technophobes. I think there is room to bridge the gap here, where technology should be utilised as a tool to make life easier, not to scare people away. Free software just by itself is probably not going to be useful for social activists in this context.

However, we believe that free software, along with the support of a strong community or guaranteed commercial support and training, would certainly be the strongest candidate to meet their technology/IT needs.

Q. Would you like to share any interesting experience your team had during the walk, with our readers?



Kerala IT secretary Ajaykumar IAS, inaugurating the felicitation ceremony for the Freedom Walkers.

There were quite a few interesting experiences. We were amazed at the way the police dealt with us as

members of the public—very professionally! We were also amazed by the hospitality of the citizens of the state. There was a reasonable balance between nasty and really pleasant experiences. However, the walk did bring to our attention a few areas where things can improve; for example, traffic management on congested roads and sufficient spacing on the sides of highways for pedestrians. Getting rid of rubbish on the sides of the road and other public places is another disgusting habit that Kerala has somehow discovered. The roadsides are littered with rubbish tossed out of passing vehicles, both private and commercial. We have been maintaining a daily blog of our experiences, with about 9,000 pictures at www.freedomwalk.in.



Q. Now that the one-month long walk is over, what are the future plans?

The main fallout from this event has been the coming together of the free software community across Kerala. Individuals from various user groups in Kerala were compelled to interact with others from other user groups, and the network has been strengthened. We also touched base with a few people working in key social and environmental areas. Our vision is that these new interactions will rejuvenate the user groups and help transform the community into one of innovators and contributors, thereby becoming indirect agents of change by perhaps supporting the non-technical community that works towards social change. Ideally, this would be a great way “to be the change they wish to see in the world.”

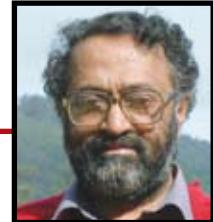
Additionally, we are planning to strategically organise workshops and introductory seminars in colleges across Kerala with the help of all the user groups. 

A BRIEF PROFILE OF THE FREEDOM WALKERS

Anoop Johnson and Cherry George Mathew are entrepreneurs running an IT firm called Zyxware Technologies [zyxware.com] based out of Trivandrum. Prasad S. R. is a freelance system integrator working in Trivandrum. Sooraj K. is working as a faculty in Ascent Engineers [ascentengineers.org], Kozhikode.

Anoop and his team have been keeping a report on a daily basis about the experiences, places visited, and photos in the freedom walk website at freedomwalk.in. The freedom walk was organised by Zyxware Technologies in association with the GNU/Linux Users Group of Trivandrum, Ascent Engineers, Swathanthra Malayalam Computing, SPACE, Free Software Users Group Calicut and Free Software Foundation of India.

By: Santhosh Thottingal is a FLOSS activist and developer. He is the project lead of Dhvani TTS project, winner of FOSS India Award 2008 and project admin of Swathanthra Malayalam Computing. He is interested in Indic language computing and blogs at santhoshr.livejournal.com.



KENNETH GONSALVES

How To Grow the Indian FOSS Movement

Everyone who has anything to do with FOSS in India is interested in growing the Indian FOSS movement. There are two theories on how to do this.

The trickle-down theory—previously known as the White Man's burden, originated in the colonial days when the white masters felt it incumbent on them to educate and uplift the natives. The idea was to select a few natives and educate them to be 'brown Englishmen'—Indian in colour, but English in thought and habit. The knowledge imparted to this elite few would then 'trickle down' to the unwashed masses.

Translated to FOSS, this implies identifying an elite few FOSS 'super stars', publicising their achievements and then watching the effect trickle down to the masses. This further implies dividing the tasks before the community into things that *real* programmers do and things that are beneath the dignity of *real* programmers. For example, translations, documentation, bug fixing and the like are not tasks for super stars. This, of course, also creates a 'caste system' among FOSS contributors, some being superior to others. So far, the attempt to grow superstars seems to have resulted in the superstars either leaving the country or becoming too swollen headed to interact with the ordinary contributors.

Who becomes a FOSS contributor? It is well known that FOSS contributors are those with access to a computer and

Internet in their spare time, and enjoy using that spare time to contribute. The contribution may be kernel code, string translation, documentation, sitting on mailing lists and IRC to answer questions, or moving around evangelising and training—or even just talking about and publicising FOSS.

So who *are* those with access to a computer and Internet in their spare time? In any non-elite college, maybe one in a 100 or, unfortunately, one in a 1000. Even in engineering colleges with good labs, the buses leave at 5 pm and only hostellites can stay beyond that and use the resources. It is that simple: the potential pool of FOSS contributors is small, so the *actual* pool of contributors is even smaller.

Further, my experience of the past two years has shown that the elite colleges in the metros are the last place where one can expect FOSS contributors. The bulk are from the second rung of engineering colleges, both private and government, in the two-tier cities and beyond. Kolhapur, Yamunanagar, Meerut, Madurai, Coimbatore, Calicut, Durgapur, districts around Chennai—these are the developing hotspots of FOSS activity. There is a great

thirst for knowledge in these areas, and a wide acceptance of FOSS, which one does not find in the more cynical audiences in elite institutions in the metros.

Another significant development is that a substantial number of FOSS contributors are emerging outside the traditional LUG infrastructure. Those LUGs who continue to do what LUGs do best—support and propagate Linux, in particular, and FOSS in general are still flourishing. Examples are the PLUG in Pune and ILUGC in Chennai. These continue to grow, are very active and are bringing in a lot of new blood. Other LUGs that have drifted away from traditional LUG activities are stagnating and have lost their way.

We now see a large number of contributors to FOSS who use Windows or OSX. Many of them become involved in the movement by contributing to free content generating sites like Wikipedia and OpenStreetMap.

OpenStreetMap is a case to the point where the online potlatch map editing

Contribute in some way or the other without worrying too much about becoming a superstar or where you are in the FOSS food chain.

application is written in Flash, which is in no way free software. But the application *is* free software—and qualifies as FOSS. Barcamps, OSCamps and similar events are producing contributors to free content and free code—most of them have never heard of RMS, the GPL or BSD, or maybe even Linux!

I personally do not worry much. As the pool of potential contributors grows, so will contributions. The only way to help out is to make sure awareness is spread—the biggest stumbling block is a lack of awareness. In every seminar and workshop I have held over the past two years, I have seen the stunned and awed expressions on the faces of people who are given a taste of FOSS—that is all we need to do, give more and more people a taste of FOSS; they will do the rest. And, of course, contribute in some way or the other without worrying too much about becoming a superstar or where you are in the FOSS food chain. FOSS is fun! If it is not fun, do not do it! 

Kenneth Gonsalves works with NRC-FOSS at AU-KBC, MIT, Chennai. He can be reached at lawgon@au-kbc.org

GUEST COLUMN

Put Some Colour on that Terminal!



Are you bored of looking at that black and white terminal output? Let's give it some colour. The xterm terminal supports colourful letters. You only need a basic understanding of the Escape sequence, and a little knowledge of its syntax.

Before we get started with the fun part, let's get some of the basics right! In addition to the capability to do many important things like moving cursor positions, printing new lines, etc, Escape sequences form the core of printing colourful text on a terminal prompt. There's a wide range of Escape characters defined for the Linux terminals -- for example, `\n` for new line, `\b` for backspace, etc.

To get started, we need the Octal Escape sequence `\033` for printing colourful words. After encountering the Escape character, the console looks for the instruction and acts immediately, based on the instruction and its parameter(s).

Although there are a set of instructions, and each instruction can have many different parameters, we will focus on CSI (Control Sequence Introducer) instructions. CSI (represented by 't') looks for parameters or a group of parameters. The parameters are normally a set of decimal letters. When we have a group of parameters, it should be separated with a semi-colon `;`.

The action of the CSI sequence is regulated by the end character. For our purposes, the end character will be `m` (lower-case M), which is responsible for the character display attributes. So that's enough with the theory part; let's now

get started with some practical examples of putting colour on that dull terminal.

Note: The article has been written taking the xterm terminal and Bash shell into consideration.

Changing the foreground colour

Try the following arguments with the `echo` command:

```
echo -e "\033[1;32m Linux, the great! \033[37m"
```

This will print "Linux, the great!" in green.

Let us decrypt the arguments of the echo statement:

- `-e` will enable interpretation of the

TABLE 1: DECIMAL CODES FOR FOREGROUND COLOURS

Colour	Code
Black	30
Red	31
Green	32
Brown	33
Blue	34
Purple	35
Cyan	36
White	37

backslash-escaped characters.

- `\033` is the code for the Escape character; encountering this, the console moves to escape mode.
- `/` is the instruction to the above Escape sequence to switch to the Command Sequence Introducer (CSI) mode.

Now CSI will look for a set of digital characters. As I have mentioned earlier, we can give multiple parameters separated by commas. `1` is the first parameter after CSI, which tells the console to print the letters in bold format. The following are a few other options:

- `0` – reset to default
- `1` – bold
- `2` – half bright
- `4` – underscore
- `5` – blink

If we don't give the above parameter, the printing will be in the default format.

Next is the main part—printing the characters in colour. Table 1 shows the decimal codes that regulate the foreground colour of the characters.

As we have given the number `32`, the characters will be printed in green.

`m` at the end sets the character attributes as per the above mentioned codes.

Finally, I used codes again at the end so as to restore the default colour (which is white in my case!).

Changing the background colour

This is as easy as foreground colour change; simply substitute the colour codes from `30-37` to `40-47` and you are ready to go. Table 2 shows the code for background colours.

Colouring the bash prompt

The main bash prompt is stored in the `PS1` variable. The following are the special characters that are decoded to their respective meanings in the typical Fedora prompt variable:

- `\u` = Username;
- `\h` = Hostname;
- `\w` = Present Working Dir;

To simplify matters, let us store codes in variables:

- `RED="\[\033[1;31m\]"`
- `CYAN="\[\033[1;36m\]"`
- `WHITE="\[\033[1;37m\]"`

Finally, the `PS1` will look like what follows:

```
PS1="[$(CYAN)\u$(RED)@$CYAN]\H:$RED\w$(WHITE)] "
```

Colouring the `ls` output

Many of us are aware of the `ls -color`, to print the file listing in the colour.

After understanding colour codes we can customise the listing colour as per our choice.

`dircolors` is the command that is used to set the `LS_COLOR` variable, which in turn regulates the colour output of the `ls` command.

Type `dircolors -p` to print the default colour listing.

TABLE 2: DECIMAL CODES FOR BACKGROUND COLOURS

Colour	Code
Black	40
Red	41
Green	42
Brown	43
Blue	44
Purple	45
Cyan	46
White	47

Here you will see variables like `FILE`, `DIR`, `EXEC`, etc, with some colour codes attached to them. Also, there are entries to specify separate colour codes for files with particular extensions.

In order to customise things, we need to modify the respective entries. First, store the colour database in one file as follows:

```
dircolors -p > a.colors
```

Now, open the `a.colors` file and modify the entries as you prefer. For example, to add the listing of Perl files in blue, you will put: `.pl 01;34`. And to have underlined output of the executable, I will modify `EXEC` to `EXEC 04;31`

After doing the changes, save the file. Now type:

```
'eval dircolors -b a.colors'
```

...and you are ready to go.

In order to make these colour settings permanent, put the above command in your login script.

Just like `LS_COLOR`, we can also have output of `grep` in colour. Type:

```
export GREP_COLOR="02;35"
```

That's about it; we are done with the basic knowledge of a colourful world. So, go ahead and put some colour on your terminal. 

NOTES AND REFERENCES:

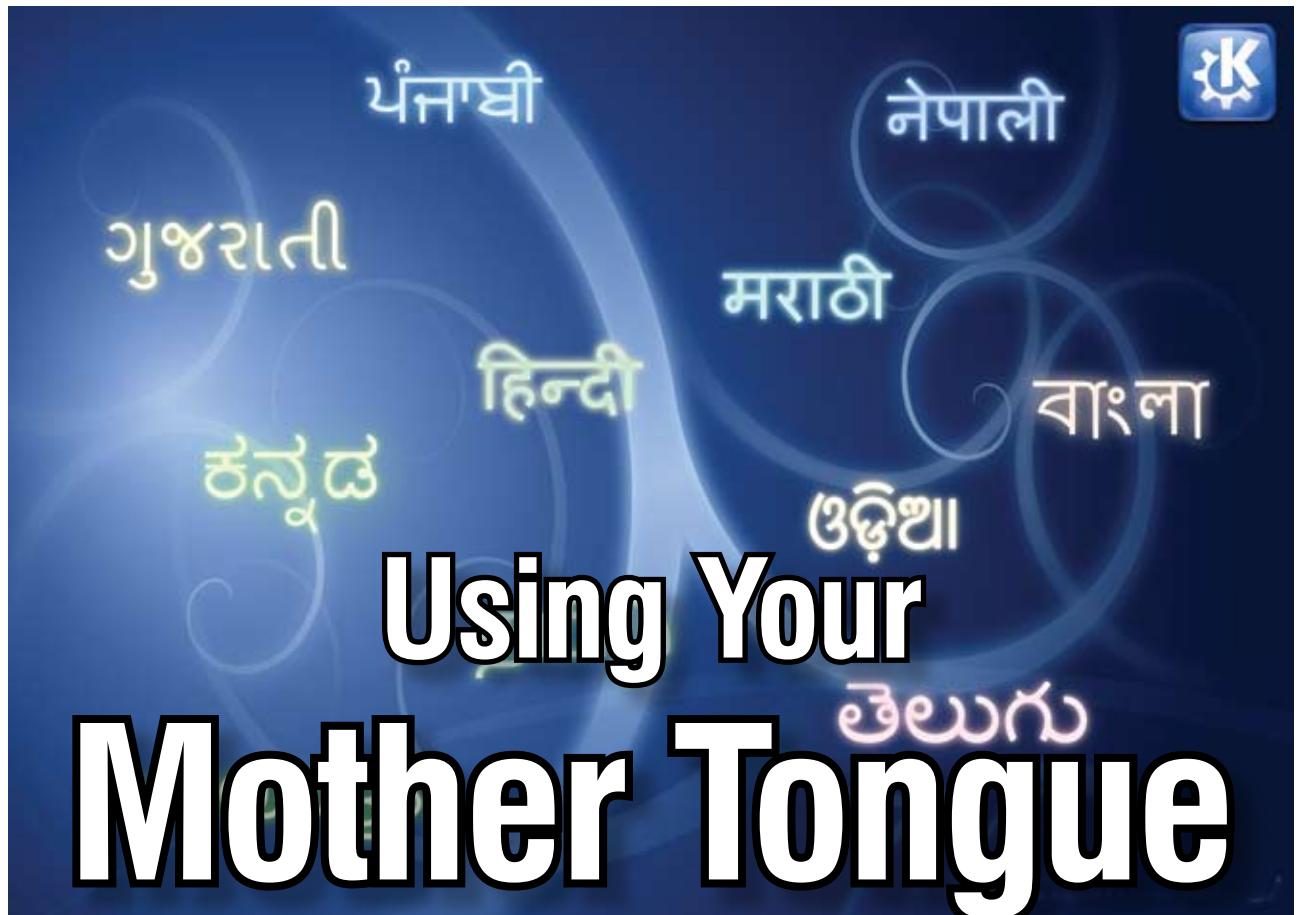
- Please refer to Edward Moy's research paper for the list of all the control sequences and its uses: netmirror.org/mirror/xfree86.org/4.4.0/doc/PDF/ctlseqs.pdf
- The list of all the colour codes can be found using the `dircolor -p` command.

By: Purohit Bhargav. The author has a deep interest in open source and is currently developing applications using Perl on UNIX/Linux. He lives in Mumbai and can be reached at purohit_bhargav@rediffmail.com



How To

For U & Me



Using Your Mother Tongue on the FOSS Desktop

Part I: It's Easy with KDE

It's time computer users could feel at home on their desktops...being able to slip into using their mother tongues, where earlier, English ruled the roost. Find out more about breaking the language barrier on the FOSS desktop.

In this country, many computer users don't find it easy to use their native languages on their Windows desktops. There is the simple matter of paying through your nose to get a licensed copy of proprietary software like Akruti, which often requires RAM upgrades if it is to work properly; or else, there is the hassle of finding and installing proprietary fonts, and learning to use them; or of installing the Baraha word processor, a no-cost, simple and closed source deal. Thanks, but not for me—not after finding out that there's considerable support for many Indian languages on the FOSS desktop.

That's right. For the past few years, there's been a healthy volunteer-led emphasis on enabling regional languages on KDE and GNOME. Indian languages have benefited from the attention too. Today, at least nine Indian scripts can be typed on the FOSS desktop; into many of these, the desktop

interfaces have been translated, to some extent. More languages are being supported by the year. Therefore, it's safe to say that most computer users will find their bilingual needs satisfied on the FOSS desktops.

There are two kinds of Indian-language 'support': the kind where you can word-process in your script; and two, where your desktop, including menus, warning messages and applications, are translated into your language. This article, Part 1 of a series, deals with word-processing on KDE. Another will deal with GNOME. A third will talk of translated desktops.

The ka-kha-ga...

It all begins with that keyboard on your computer, doesn't it? Q-W-E-R-T-Y-U-I-O. It comprises the English alphabet. This is the same keyboard you'll have to use to type in Indian

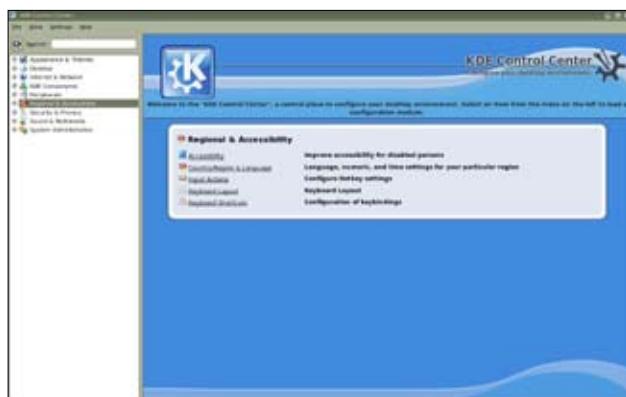


Figure 1: The 'Regional & Accessibility' group of options in KControl

languages. But how do you do that? This is how: you'll have to use an application called the 'keyboard layout changer'. It instructs the computer how to produce different letters when you press particular keys.

Today, we look at KDE's layout changer. The KDE Keyboard Tool is installed with KDE by default, and ships with many Indian-language layouts. So, languages can usually be enabled in KDE without any downloading involved, and with very few clicks. In my KDE 3.5.9 (Mandriva 2008.1), there are four Tamil layouts, three Hindi layouts (including OLPC), two layouts each for Bengali and Malayalam, and one layout each for Kannada, Gujarati, Oriya, Telugu and Punjabi. (But, sorry, Marathi users -- use the OLPC layout; that's right, the *dnaya*



Figure 2: The 'Keyboard Layout' section in KControl

and *Lla* alphabets are not provided elsewhere. Yes, it's the year 2008. Any Marathi font specialists reading this, please do help.)

So, as you can see, many of the Indian language layouts are covered.

Changing the layout in KDE

The KDE Keyboard Tool is invoked through the KDE Control Centre. Launch it, and you'll see the 'Regional and Accessibility' group of options (Figure 1). In that, select the 'Keyboard Layout' section (Figure 2). Now you'll be confronted with a pane. Here, ensure that 'Enable Keyboard Layouts' is selected. You should now see two columns—'Available Layouts' and 'Active Layouts'. Below, you have the 'Add' and

www.eleb2b.com

Electronic Components Products Machinery

Will now be sold here

Presenting Electronics Bazaar, India's First B2B Sourcing Guide For Electronics Components, Products and Machinery.

Unit Price: Rs 50
Subscription Rate: Rs 350 (12 Issues)

EFY Enterprises Pvt Ltd, D-87/1, Okhla Industrial Area, Phase-I, New Delhi 110 020; Ph: 011-26810601-03; Fax: 26817563; Email: info@efyindia.com; Website: eleb2b.com

Looking for LINUX, UNIX, OSS Training & Services ?

- # IT Infrastructure Services
- # Web Presence Services
- # Technical Training
- # SysAdmin Staffing

Our Partnerships:



tuxg **The UX Group**
A SysAdmin Company

40-1-92, Khanna Nagar

Benz Circle, VIJAYAWADA-10

✉ 0866-6662368 ☎ 9989777150

<http://theuxgroup.com>

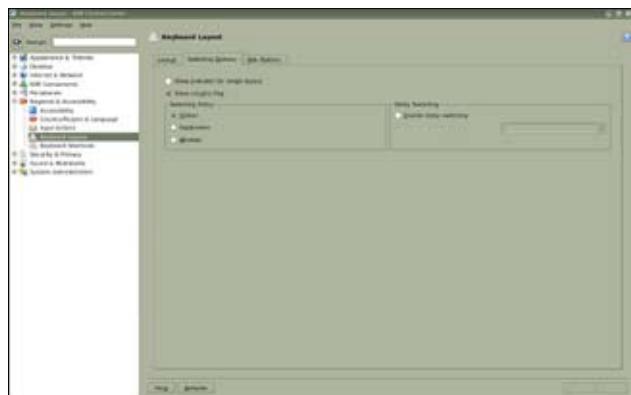


Figure 3: 'Switching Options' to set language inputs globally

'Remove' options. Pretty self-explanatory, isn't it? Below that, you can choose a 'Layout variant' -- phonetic or non-phonetic. (Devanagari users will need to select the Bolnagri layout for phonetic use, and perhaps Remington among the non-phonetic choices.) A few words on phonetic layouts later.

Okay, that's done! You've selected your languages of choice, but you can configure the keyboard tool further still. In the Control Centre, you should see three tabs at the top of the window. Choose the 'Switching Options' tab (Figure 3) from these. Now you'll have the option of changing the language input globally. This will enable you to type in your preferred language globally across applications. There are alternatives to this 'global' option: in the same pane, you can make the language change apply to selected windows only. The choice depends on your style.

Now, if you'll just take a look at your desktop's main panel (usually it's at the bottom of the screen), close to its right corner you'll see the embedded icon of the KDE Keyboard Tool. Typically, this icon is the flag of the country whose language is selected. Right-click and select your language from the menu; the flag icon will change accordingly. Easy?

The tool recognises keyboard shortcuts too; you can customise one. You might need a custom shortcut in some distros, though, because the default shortcut sometimes clashes with the 'change desktop' shortcut if you've enabled Compiz Fusion. Personally, I just use the mouse-click method, which takes a few seconds more, since I don't need to change the layout on-the-fly.

In KDE 3.5, the keyboard layout application is very stable across distros. For me, it hasn't crashed even once in years, on various platforms.

Phonetic and non-phonetic keyboard layouts

There are the two types of layouts provided in KDE. Both are radically different plans of linking alphabets to keys, aimed at newbies and pros, respectively.

Phonetic, which means 'related to sound', is the type of layout that links the keyboard's keys to similar-sounding alphabets. With Hindi, for example, a phonetic layout means that Key P will produce the Devanagari letter *pa*. Key N produces *na*. And so on for all the keys. There it is now, your alphabet distributed according to the QWERTY pattern. Very

intuitive, isn't it; something new users would take to. The only thing is, the phonetic layout is not ergonomic. You might find yourself stretching your finger to the edge of the keypad for a letter in frequent use. It's a strain during long typing sessions.

Which brings us to the non-phonetic layout. It links the keys to the alphabets according to ergonomic considerations: their frequency of use, and the reach of your fingers. That is why these layouts are considerably faster to type in than the phonetic ones. Yes, they can be a bit counter-intuitive. For example, in the non-phonetic Hindi layout, OLPC, Key P produces the letter *ja*. And so on. Presumably, it's a more ergonomic location for that letter, set according to its frequency of use. You'll have to spend some time in learning the layout. But once you do, it's a fair bet to say you won't use the phonetic version again.

A word on saving Indian language documents

Now that you've keyed in text, what format do you save it in? That depends! If your intended reader uses any mainstream FOSS desktop (except for some on KDE4: the desktop may not carry the required fonts by default, and you'll have to download them from your distro's repository), go right ahead and save in whichever format you like; preferably the OpenDocument format. But you're better off using PDF if you're not sure your recipient has the fonts needed, or if you're sending your document to someone on, dear Lord, Windows. They can still read your PDF document if they don't have the fonts.

There is the Unicode text format also, but it is not universally implemented. PDF is a stopgap solution until a popular, universal format emerges for Indian languages across platforms, like plain text for English. The popular applications KWrite, OpenOffice.org or Abiword have inbuilt PDF conversion. (Make sure you print to PDF with the 'embed fonts' option turned on.)

A few last words

Indian languages on the FOSS desktop are a viable proposition. As we've seen, they're easy to set up on KDE3.5.x. All that remains to be done is to spread the word around, and most people need not look at expensive and proprietary solutions any more. 

By: Suhit Kelkar is a freelance journalist and translator based in Mumbai. He can be contacted on suhitkelkar@gmail.com

The main illustration of this article is a copyright of Kamaleshwar Morjal, licenced under Creative Commons Non-Commercial Share Alike Licence 2.0, and is hosted at www.flickr.com/photos/anuragp/3039862173. The copyright holder has granted LFY the permission to use this image for the commercial purpose of publishing and selling this article, and it should not be treated as part of this article.

The image is part of the KDE posters for 2008 winter collection [www.flickr.com/photos/anuragp/sets/72157609058551029]. It showcases all the official Indian languages (Bengali, Gujarati, Hindi, Kannada, Malayalam, Marathi, Nepali, Oriya, Punjabi, Tamil, Telugu) in which KDE is being translated into and has an entry in the KDE localisation website.



When a Desi-crafted Card Meets Software... PIAF!

PIAF, or PBX in a Flash, likes to call itself "the lean, mean Asterisk machine." Is it really so?



Dinesh Birla wrote in from Singapore. We had not met, but just encountered one another via the India-GII mailing list, a useful place for people discussing tech matters... or writers focusing on it.

Dinesh wrote: "I am basically looking for someone to do a product review on a VoIP product that is in the market today."

He pointed to www.voxzone.com and said: "It's a VoIP entry card aimed at hobby VoIP enthusiasts, who want to try VoIP and get familiar with Asterisk systems. It is hardware-based. It is also my product. It works with variants of Asterisk systems that are based on Digium's free open source PABX system."

Asterisk, as you know, is a software PBX that connects fixed lines and Internet telephony. Birla wrote, "Yeah, it works on [GNU]Linux. There is a pbxinaflash CD I ship it out with, also. It's an open source PBX, and converts your machine into a PABX."

This drew my interest. A few e-mails later, a small packet arrived in the post. It contained... the promised card.

This VoIP entry card is known as Voxzone X100P.

Says Dinesh Birla: "The card is known to work globally, supporting different telephone carrier networks."

We (a techie friend who's word I respect) tried it and got stuck the first time. Something wrong? Card damaged?

Then, I passed it on to Ricky. Though a Windows type of guy, he was immediately interested. He promptly

accepted the “try it and see” offer.

A few days later, he wrote back to say, “We completed the review of the Voxzone X100P Card. Found it pretty good actually, and may probably even consider building such systems (Linux or Win32) for commercial or business applications.” Ricky is part of Online [www.opspl.com] in South Goa.

Finally, it reached the Margao-based networking and support engineer Francisco Miranda. Cisco, as he is widely known, used this as an opportunity to also test out the PIAF (PBX In A Flash) software with the card.

Voxzone for Asterisk

The Voxzone X100P is a single FXO interface meant to connect the Asterisk PBX server to the PSTN (Public Switched Telephone Network). Voxzone X100P's techies say the card is 100 per cent compatible with the Digium X100P and the Asterisk PBX.

Using the Voxzone X100P FXO Card and the Asterisk PBX—the manufacturers say—one can easily deploy such services like callback (need two cards), calling card business, VoIP gateways and IVR.

To try and do this, we used the Voxzone X100P FXO PCI card, sent in by Dinesh Birla. This came in a box.

To set it up for an Asterisk PBX server, we obtained a copy of PIAF (PBX In A Flash), which we set up, with the Asterisk PBX.

The following is a review of the total system:

- The Voxzone X100P FXO PCI CARD for Asterisk
- The PIAF (PBX In A Flash) System with the Asterisk PBX.

PIAF first!

PIAF calls itself “the lean, mean Asterisk machine.” You can find out more about it at pbxinaflash.com.

Here's a description of the project: “If you've longed for the good ol' days of Asterisk@Home, welcome back to the new steroid-enhanced version. PBX in a Flash is the lean, mean Asterisk machine designed to meet the needs of hobbyists as well as business users and VARs.

“You'll have a high-performance turnkey Asterisk PBX that's easy to upgrade with dozens of add-on scripts to provide virtually any feature you can imagine. And you can choose from tons of Nerd Vittles and FreePBX applications that install in under 15 seconds: AsteriDex, weather reports, news feeds, e-mail by phone, telephone reminders, and many more. You add features when you need additional functionality. Otherwise, just say no to bloatware!”

In other words, PIAF is a GNU/Linux-based distribution that will turn your PC into a free ‘Private Branch Exchange’ allowing you to make phone calls via VoIP or PSTN trunks from all your internal SIP IP-Phones or computer-based soft-phones.

PIAF is a standardised implementation of Asterisk and is based around a Web-based configuration interface (Webmin) and other tools.



Contents of the Voxzone Internal PCI Foreign Exchange Office (FXO) hardware package as supplied by Voxzone. Also included is the FreePBX software CD-ROM with the linux operating system (not displayed).

The following are some of its features:

- You can add or change extension and voice mail accounts in seconds
- Native support of SIP, IAX, and ZAP clients (other endpoints are supported through custom extensions)
- Supports all Asterisk-supported trunk technologies
- Modular, with an online repository to add/upgrade features in the interface
- Reduces long distance costs with LCR and powerful pattern-based outbound routing
- Routes incoming calls based on time-of-day, DID, Caller ID
- Creates interactive digital receptionist (IVR) menus
- Designs sophisticated call groups
- Enables personalised find-me/follow-me
- Manages callers and implements call centres with queues
- Uploads custom on-hold music (MOH)
- Searches company directory, based on first or last name
- Detects and receives incoming faxes
- Shares administrative duties
- Backs up and restores your system
- Saves audio recordings of calls
- Views call detail reporting with asterisk-stat
- Views extension and trunk status with Flash Operator Panel
- Views conversation recordings with Asterisk Recording

Interface (ARI)

Cisco had to search the Internet and download PIAF and its user guides since they were not provided in the CD-ROM.

The installation is quite straightforward; insert the CD-ROM into the drive and set the BIOS to boot from CD-ROM first, and then the hard disk.

When the CD-ROM boots-up, you are offered Asterisk 1.4 or Asterisk 1.6 Beta for installation, along with various options. We decided to install Asterisk 1.4 since it was recommended for production set-ups. Cisco chose a normal install with LVM.

After the operating system was installed and the system

rebooted, Cisco was offered the option to download the latest, or install the 'payload' file from the CD-ROM. He chose to download the latest version from the Internet, as recommended.

With the installation complete, he next had to log in as the root to carry out a few more recommended steps to start using PIAF:

1. Update the scripts by running *update-scripts*. Do a *help-pbx* after the update finishes and you will see all of the new programs that you can use.
2. Run *update-fixes* to update PBX in a Flash with any patches that did not make it into the current release of PIAF.
3. Run *passwd-master* to set most passwords in PIAF. Please note: null passwords are not allowed.
4. Run *netconfig* to configure the network interface to use a static IP address.
5. Reboot the system.
6. Edit 'zaptel' in */etc/sysconfig*, and comment-out the unused hardware with # marks.
7. Run *genzaptelconfig -d -v -s -z* to correctly configure zaptel for our hardware (FXO card).
8. Run *zttool* to confirm hardware installation of FXO card.
9. Reboot the system.

Once Cisco was done configuring the operating system, Asterisk and hardware set-ups, FreePBX could be accessed and further configured from the Web interface, which was found at *http://(IP configured in step 4 above)*.

Once at the Web interface, we completed just a few steps to get the FreePBX system running:

1. Enable the *configedit* module in module admin.
2. In *tools-->configedit*, edit the file *Zapata.conf* and add the line *puledial=yes*.
3. Configure the extensions. (SIP soft-phones.)
4. Configure options on the general page of FreePBX administration.
5. Configure the out-bound route (PSTN provider, in our case) on the out-bound routes page, including dialling pattern.
6. Configure the trunk on the trunk page.
7. Configure incoming calls to go to a specific soft phone or extension.
8. Reboot the system.

All that remained to be done was to download a soft phone of choice, and then install and configure it on the required clients. And then we were ready to go!

Overall, Cisco says he found PIAF to be easy to install. The installation was quite quick with a broadband connection. Configuration was relatively simple.

Unfortunately, most of the time was wasted on trying to figure out how to get pulse-dialling to work, since at first we would get "all lines in this route are busy" messages whenever we tried to dial-out to PSTN.

We did not use a FXS card because we did not use any analogue phones to connect to the PBX; only soft-phones were used.

Configuring a supported FXS card should be easy anyway. Using this set-up we were able to call any other soft phone in the office and vice-versa. We were also able to call mobile phone numbers as well as PSTN landlines from our soft phones.

For our example, we used the free soft phone from Counter Path called x-lite. This was installed on the desired Windows XP clients. You have to dial ***7469 (SEND) to bring up the x-lite advanced configuration window. Now, filter for 'honor', and double click the 'honor' entry to change the value to '1'.

Cisco says he found the PIAF Asterisk 1.4 to be quite stable and decent enough for production use. "The Voxzone X100P used in this test is a single port internal PCI FXO card that uses the zaptel driver. This card was detected and easily configured using the *genzaptelconfig* command described earlier," he said.

For this test, a 32-bit processor was used, but 64-bit is also supported. There are reports of significant performance improvements on 64-bit processors using the 64-bit version of the software.

Server requirements

The recommended server requirements are as follows:

- CentOS 5.x (installed along with the CD-ROM set-up).
- Pentium 4 or equivalent processor. 64-bit is also supported.
- 2 GB RAM. We ran the test set-up with 512 MB and a couple of soft phones satisfactorily, but 2 GB is recommended for production use.
- 80 GB HDD or better. Both SATA and IDE are supported
- CD-ROM drive
- FXO/FXS card/s

The following is the hardware configuration of our test rig:

- Pentium 4 @ 1.8GHz
- 512MB DDR2
- Intel 945 motherboard with integrated VGA
- RTL8139 NIC
- 160GB SATA HDD
- 52x CDROM Drive
- Keyboard + mouse
- Voxzone X100P FXO (PCI internal 1-Port card).

Cisco concluded: "Quite a nice system. It set up pretty easily and worked satisfactorily."

A nice experience, overall... getting to know someone through the Internet, learning of their products, and in turn, experimenting with more software to get things to work. And work, it does! 

By: Frederick Noronha and Francisco (Cisco) Miranda.

Fred is a journalist based in Goa, and a non-techie deeply committed to Free Software-using it, promoting it, sharing its power with others. Cisco is a networking and support engineer affiliated with Online Productivity Solutions (www.opspl.com). You can reach them at fn@goa-india.org and cisco@opspl.com, respectively.

17th Convergence India 2009

International Exhibition & Conference

Pragati Maidan, New Delhi, India

18-20 March 2009

*South Asia's largest ICT Event
Digital Convergence changing the ICT Landscape*



Featuring

- Telecom
- Broadcast
- Mobility
- Cable
- Information Technology
- Satellite
- Information Security

Certified by



Department of Telecommunications
Ministry of Communications & Information Technology
Government of India

Supported by



Department of Information Technology
Ministry of Communications & Information Technology
Government of India



Ministry of Information & Broadcasting
Government of India



CEAMA

COFI

ELCINA

CELLULAR ASSOCIATION
ICA

IFCII

IP TV
INDIA FORUM

ISPAI

ITSMA

MAIT

SNIA

TEMA

Wi-Fi
Alliance

Supporting Journal



Organised by



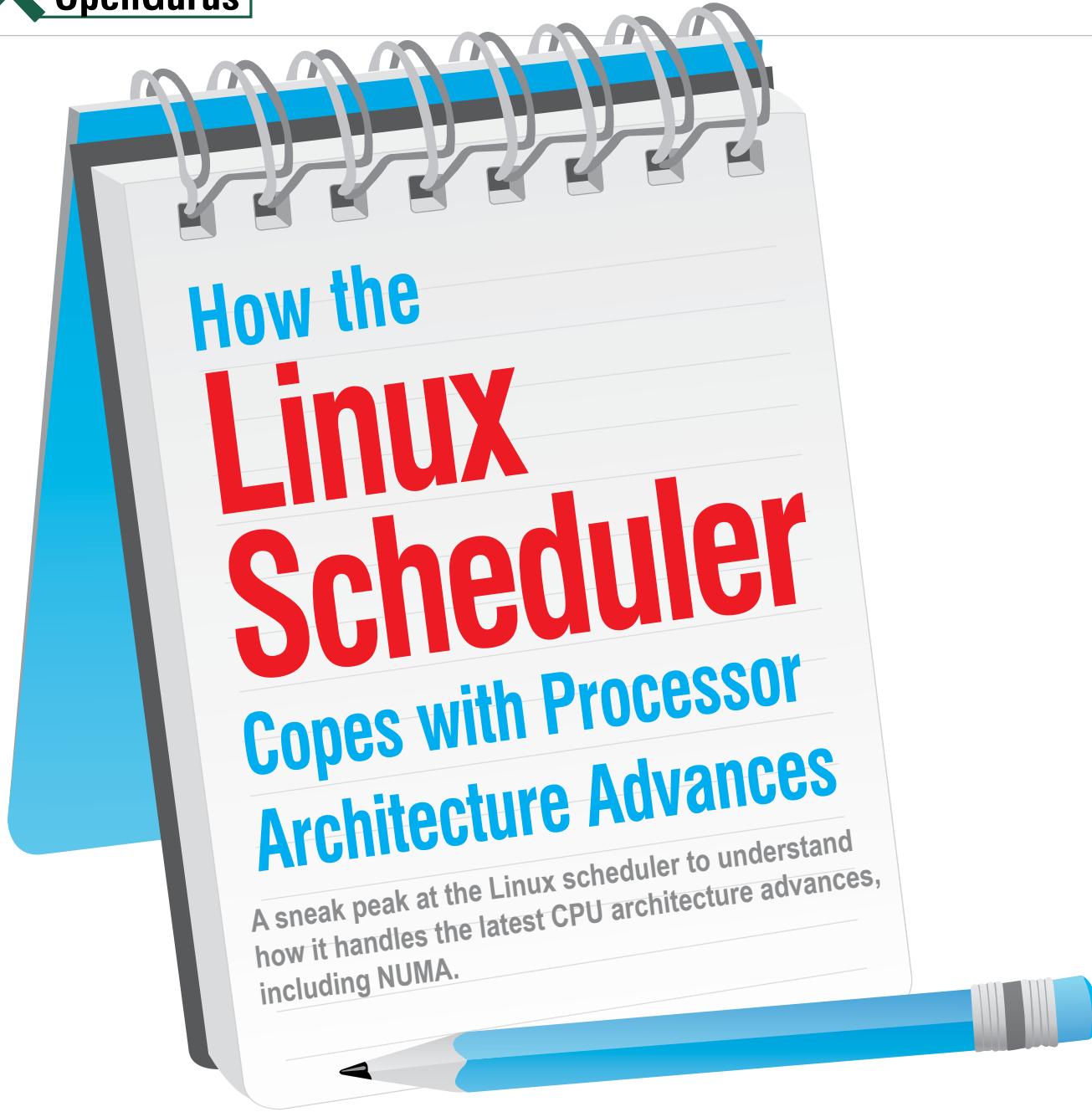
Exhibitions India Pvt. Ltd.
(An ISO 9001:2000 Certified Company)

217-B, (2nd Floor) Okhla Industrial Estate, Phase III, New Delhi 110 020, India Tel: +91 11 4279 5000 Fax: +91 11 4279 5098/99

Bunny Sidhu, Vice President, (M) +91 98733 43925 bunnys@eigroup.in /
Sambit Mund, Group Manager, (M) +91 93126 55071; sambitm@eigroup.in

Branches: Bangalore, Chennai, Hyderabad, Mumbai, Ahmedabad, California

www.convergenceindia.org



How the Linux Scheduler Copes with Processor Architecture Advances

A sneak peak at the Linux scheduler to understand how it handles the latest CPU architecture advances, including NUMA.

Multi-tasking or multi-processing has been an inevitable feature of operating systems since their inception. Multi-tasking provides the OS a capability to run multiple processes—although, one by one, yet appearing simultaneous. The component that makes the OS capable of multi-tasking is known as a scheduler. The scheduler has the responsibility of determining if a process should or shouldn't continue to run, or which processes should run next and on which CPU.

While a scheduler can be as simple as selecting all the ready-to-run processes in round-robin order, modern OSs use a very complex algorithm to ensure fairness to all the processes, while maintaining optimum CPU utilisation; and Linux is no exception. The scheduler has two prime responsibilities: fairness among processes and load balancing among CPUs. While the Linux scheduler uses a complex algorithm and heuristics to determine the nature of process (IO bound, CPU bound, etc), this article focuses only on the load-balancing

aspect. Therefore, the term 'scheduling', in this article, should be assumed as a scheduler activity to reduce load imbalance among CPUs. The scheduler does it by moving processes from busy to idle CPUs, yet not compromising with system performance.

Now, with the advent of more sophisticated and improved hardware fabrication technologies, CPU architectures have gone through radical changes. Modern high-end systems are SMP-enabled, where more than one CPU shares the processing. This was made more complex by hyper threading, introduced by Intel, where a single processor can run more than one process at a time.

Since a high-end system has multiple CPUs sharing common resources (memory, the bus, etc), the common resources become a bottleneck in performance.

To eliminate this bottleneck, another architecture improvement, popularly known as NUMA, was introduced. NUMA architecture allows a subset of CPUs to have faster access exclusively to certain resources. Lastly, multi-core architectures are making a mark in new systems.

While these architecture innovations showed great potential to improve system performance, they posed a big challenge to the scheduler because a scheduling decision has to be made intelligently to satisfy the various requirements of the architecture. Before we move forward to understand how the Linux scheduler dealt with these architectures, it's time to look into some details of the above-mentioned architectures and their requirements, with respect to the scheduler.

A brief introduction to hyper threading

Hyper Threading Technology (HTT) was introduced by Intel to improve the parallelisation of computation. Hyper threading is done by duplicating certain parts of the processor (control registers, general purpose registers, etc) that maintain the architectural state while not duplicating the main execution resources. Therefore, another process can be scheduled on the HT-enabled processor when the processor is halted for cache miss, branch misprediction, etc.

The scheduler considers a HT-enabled CPU as two logical CPUs that can share the cache. Therefore, if a process that is cache hot (i.e., even though the process had been scheduled out, the cache is still valid) is to be scheduled, the scheduler should choose another logical CPU of the HT-enabled processor to run the process, because both logical CPUs can share the cache.

Therefore, HTT processors are also known as SMT (Symmetrical Multi Threading) processors.

A brief introduction to NUMA

Non Uniform Memory Access (NUMA) is an extended version of Symmetrical Multiprocessing Architecture (SMP), where memory access time depends upon the memory access location relative to the CPU. Therefore,

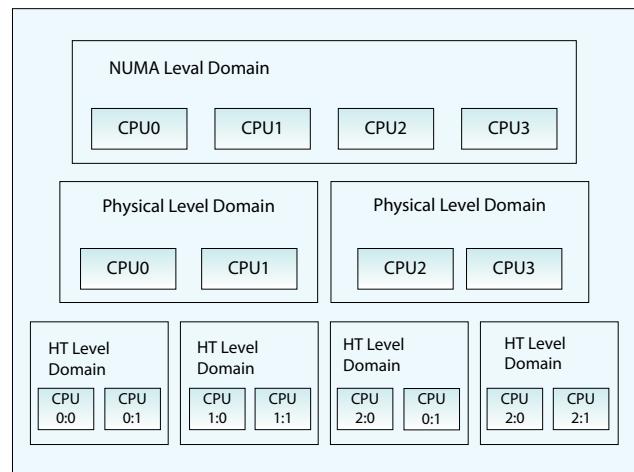


Figure 1: Scheduling domain hierarchy for a SMT NUMA machine

different processors have different access times to a certain memory location. Thus a process running on a CPU will have degraded performance if the process is moved to another CPU that has higher access time to the memory being used by the process. A scheduler must take this aspect into account before deciding to move a process from one CPU to another.

Scheduler responsibilities

Given the above description, let's consider a SMP system that has some or all HT-enabled processors on NUMA architecture. The scheduler has to take into account certain limitations, while ensuring maximum CPU utilisation and fairness. A few of them are as follows:

1. If a system is under CPU load imbalance, CPU load balancing must be done, i.e., a few processes must be moved from the busy CPU to the idle one.
2. If a process is cache hot, it should be run on the same CPU (to reuse the cache). If the CPU is HT-enabled, the second logical CPU should be considered to run the process.
3. The process should not be moved between NUMA nodes (from one subset to another where memory access is slower) until really needed.

Scheduling domains

Linux introduced a concept of scheduling domains to make the scheduler aware of the processor topology. The topology-aware scheduler is more flexible than the earlier O(1) scheduler and fulfills all the requirements discussed earlier.

The scheduling domain refers to a group of CPUs whose load can be balanced against each other. The scheduling domains are hierarchical, and load balancing is done starting from the base domain since CPUs at the bottom of the hierarchy are closely related. (For example, two logical CPUs of a HT-enabled processor that can share cache.) Load balancing is performed at lower domains more frequently than higher levels. To make

TABLE 1: LINUX SCHEDULING POLICIES

Flag	Description
SD_LOAD_BALANCE	The domain is eligible for load balancing
SD_BALANCE_NEWIDLE	Balance when the domain is about to be idle
SD_BALANCE_EXEC	Balance on exec system call
SD_BALANCE_FORK	Balance on fork, clone system call
SD_WAKE_IDLE	Wake to idle CPU on task wakeup
SD_WAKE_AFFINE	Wake task to waking CPU
SD_WAKE_BALANCE	Perform balancing at task wakeup
SD_SHARE_CPUPOWER	Domain members share CPU power
SD_POWERSAVINGS_BALANCE	Balance for power savings
SD_SHARE_PKG_RESOURCES	Domain members share CPU package resources
SD_SERIALIZE	Only a single load balancing instance

this concept comprehensible, consider an SMT NUMA machine with four HT-enabled CPUs. The four CPUs are divided into two NUMA nodes, each having two CPUs. The pictorial view of the scheduling domain hierarchy for this system is described in Figure 1.

As mentioned earlier, the scheduling group is a set of CPUs that can be balanced against each other. For example, at the HT Level domain, CPU 0:0 and CPU 0:1 can be balanced against each other. Note that CPU 0:0 and CPU 0:1 are two logical CPUs of a single HT-enabled processor, CPU 0. As mentioned earlier, Linux treats a HT-enabled CPU as two logical CPUs.

At the next higher level, i.e., at the Physical Level domain, CPU 0 and CPU 1 can be balanced against each other. Similarly, CPU 2 and CPU 3 can be balanced against each other. At the next higher level, i.e., at the NUMA Level domain, all CPUs (CPU 0, CPU 1, CPU 2 and CPU 3) can be balanced against each other.

Since process scheduling at the lowest level is less costly (CPUs can share cache), scheduling is performed more frequently and even for small imbalances. At the next higher level, scheduling is slightly costly (CPUs can share memory but not cache); the scheduling is performed at a larger interval and for higher imbalances. At an even higher level, scheduling is very costly (because CPUs can not share memory at the same speed and accessing other CPUs' memory is slow), and is therefore performed at large intervals and for high imbalances.

It's time now to look at the Linux kernel source to

know how the kernel actually performs scheduling.

Linux scheduling domains and policies

The article refers to kernel 2.6.27 to explain scheduling domain implementation and policies used for the scheduling domains. The scheduling domains' policies are controlled by a few flags described in Table 1.

A combination of the above policies is used along with different scheduling domains to fulfill different requirements of each scheduling domain. The following examples show the policy flags used by different scheduling domains:

- HT Level Scheduling Domain initialisation: As defined in the header *include/linux/topology.h*, *SD_SIBLING_INIT* initialises scheduling domains flags to:

```
#ifdef CONFIG_SCHED_SMT
#define SD_SIBLING_INIT (struct sched_domain) {
...
.flags = SD_LOAD_BALANCE \
| SD_BALANCE_NEWIDLE \
| SD_BALANCE_FORK \
| SD_BALANCE_EXEC \
| SD_WAKE_AFFINE \
| SD_WAKE_IDLE \
| SD_SHARE_CPUPOWER,
...
#endif
```

- Physical Level Scheduling Domain initialisation: As defined in the same header file *topology.h*, *SD_CPU_INIT* initialises scheduling domain flags to:

```
#ifdef CONFIG_SMP
#define SD_CPU_INIT (struct sched_domain) {
...
.flags = SD_LOAD_BALANCE \
| SD_BALANCE_NEWIDLE \
| SD_BALANCE_FORK \
| SD_BALANCE_EXEC \
| SD_WAKE_AFFINE \
| BALANCE_FOR_PKG_POWER,
...
#endif
```

- NUMA Level Scheduling Domain initialisation: Again, in the *topology.h* file, *SD_ALLNODES_INIT* initialises scheduling domain flags to:

```
#define SD_ALLNODES_INIT (struct sched_domain) {
...
.flags = SD_LOAD_BALANCE \
| SD_BALANCE_NEWIDLE \
| SD_WAKE_AFFINE \
| SD_SERIALIZE \
...
}
```

It is worth discussing a few important flags now. Every scheduling domain sets the flag *SD_LOAD_BALANCE*, i.e., every domain is eligible for load balancing. This means that although load balancing at higher domains is costlier, it is not ruled out. Similarly, every scheduling domain sets the flag *SD_BALANCE_NEWDLE*; which means that if the CPU is going to become idle, it attempts to pull processes from other CPUs, to improve processor utilisation.

However, observe that only the HT Level Scheduling Domain and Physical Level Scheduling Domain set the flag *SD_BALANCE_FORK* and *SD_BALANCE_EXEC*. Since forking or cloning refer to existing memory (parent process context, mm context, etc), it is recommended to schedule the process in the same node group.

Linux load balancing implementation

Having understood the scheduling domains and the policies, it is time to have a look at the way Linux implements CPU load balancing. Linux performs load balancing through *SCHED_SOFTIRQ softirq*. The *softirq* is installed in the *sched_init* function, as follows:

```
void __init sched_init(void)
{
...
#endif CONFIG_SMP
    open_softirq(SCHED_SOFTIRQ, run_rebalance_domains,
    NULL);
#endif
...
}
```

The *SCHED_SOFTIRQ* is raised by the *scheduler_tick* function. The *scheduler_tick* function is invoked by the tick handler code with HZ frequency. The function raises *SCHED_SOFTIRQ* if the current value of jiffies is greater than *next_balance* jiffies (set earlier) for the given CPU:

```
void scheduler_tick(void)
{
...
    if (time_after_eq(jiffies, rq->next_balance))
```

```
    raise_softirq(SCHED_SOFTIRQ);
```

```
...
```

The function *run_rebalance_domains* is invoked when *SCHED_SOFTIRQ* is raised. For all domains, it checks if load rebalancing is required and invokes the *load_balance* function to do the load balancing:

```
static void run_rebalance_domains(struct softirq_action *h)
{
...
for_each_domain(this_cpu, sd) {
    if (!(sd->flags & SD_LOAD_BALANCE))
        continue;
...
if (time_after_eq(jiffies, sd->last_balance + interval))
    if (load_balance(this_cpu, this_rq, sd, idle, &balance)) {
...
}
```

The *load_balance* function checks if a scheduling domain is highly imbalanced. It does this by calling the function *find_busiest_group* and *find_busiest_queue*. Later, the *load_balance* function invokes the *move_tasks* function to move processes from the source *runqueue* to the *local_runqueue*. This completes the Linux load balancing mechanism.

The Linux scheduler has gone through drastic changes with the advent of SMP, HTT, NUMA and multi-core architectures. To support new architectures and to be more flexible with such architectures, the scheduling domain concept was introduced. The scheduling domains, their properties and their relationships with other scheduling domains help the Linux scheduler in taking intelligent decisions to ensure maximum processor utilisation and still maintain fairness among processes. 

By: Mohan Lal Jangir is working as a development lead at Samsung India Software Operations, Bangalore. He has a master's degree in computer technology from IIT Delhi, and is keenly interested in Linux, networking and network security.

Enterprise Mail Server, Linux SBS Server, Anti SPAM, Antivirus and HTTP Filtering.

Bandwidth Management, Internet Access Control, Content Filtering, Web Access Reporting.

Customised Linux Product Development.

TechnoInfotech™

udev Unplugged!



Find out what's up with this geeky utility called *udev*, and in the process learn how to auto connect to the Internet as soon as you plug in that USB modem. Or take a back-up of your home directory to start automatically as soon as you connect an external hard drive.

You plug your back-up hard disk in! After a few seconds, you get a notification: "Back-up is complete." You then unplug the hard drive and your back-up for the day is ready with you. Now imagine this: you plug your EVDO/CDMA Internet data card in, and within a few seconds you get a notification: "Internet connected." When the device is unplugged, you get a message stating the Net is disconnected. Can you ever think of such a user experience under GNU/Linux?

Of course you can! *udev* helps you achieve this and a lot more. Let's tune into what's so great about *udev*!

What is *udev*?

udev is a device manager for Linux that runs in user space. It deals with device node creation, while taking care of the persistent naming of devices upon the availability of real hardware.

By the UNIX concept, everything is a file.

We access our devices via corresponding files in the `/dev` directory. As you know, `/dev` is a directory containing device nodes for all standard devices. Traditional UNIX systems had static device nodes under the `/dev` directory. What happens when you plug your MP3 player in the USB port?

You might have noticed that it is `/dev/sda1`, or some other node, through which you access the contents of the filesystem. `/dev/sda1` is a device node corresponding to that device. This kind of static device node system worked fine, since there were a limited number of devices in earlier times. The existence of these device nodes was independent of actual devices connected to the hardware. It was a real hassle to decide whether a piece of hardware existed or not, since all possible device nodes existed.

Now, as the number of Linux-supported devices increased, especially USB removable devices and IEEE 1394 (Firewire ports), the number of static nodes required under `/dev` increased to a huge number—nearly

18,000—and it became unmanageable. Also, if some device nodes that corresponded to a connected device did not exist under `/dev`, you had to Google for the major and minor number for the device, and create the device node manually, using the `mknod` command. Since each device has its unique major and minor numbers, this was a pretty tough situation!

As a result, a pseudo RAM-based filesystem `sysfs`, mounted under `/sys`, was introduced. Users now could check whether a device existed or not, by looking into the directory tree of devices under `/sys`. Still, this wasn't a satisfactory solution, since either of the devices were statically built, or we had to create the device nodes manually, using major and minor numbers for the corresponding device.

Give the following `tree` command a try:

```
[slynux@gnubox ~]$ tree /sys/class/
```

Since our area of interest is making life easier with `udev`, let's move on to hacking `udev`.

`udev` runs in the memory all the time as a daemon and listens to kernel messages. The kernel always sends a message whenever it notices a hardware change. You can observe it by running the `dmesg` command. The following is the `dmesg` output when I connect an external hard disk:

```
# dmesg | tail

sd 5:0:0:0: [sdb] Attached SCSI disk
sd 5:0:0:0: Attached scsi generic sg2 type 0
kjournald starting. Commit interval 5 seconds
EXT3 FS on sdb, internal journal
EXT3-fs: recovery complete.
EXT3-fs: mounted filesystem with ordered data mode.
```

So, let's take a look at the duties of `udev`:

- Listen to kernel messages. If some device is connected, create its device nodes according to the order in which it is connected. `udev` has the ability to identify each of the devices uniquely. Device nodes are created only when the device is connected.
- Removal of device nodes when the device is unplugged.
- Create symlinks for device nodes, and execute commands upon `udev` events.
- Follow the `udev` rules. The `udev` daemon is controlled by a set of user-specified rules.

Consider the following scenario, with which I'll try to elaborate the usefulness of `udev`. Let's suppose you have two printers—one an inkjet and the other a laser colour printer. Usually, the one that is connected first is designated as `/dev/lp0` and the second one `/dev/lp1`. How do you understand which one is laser and which one is inkjet? Is it by looking at which one is switched on first?

`udev` is brilliant in solving such nonsense. What if you are able to get `/dev/laser` for the laser printer and `/dev/`

`inkjet` for the inkjet printer. `udev` can identify each of the devices uniquely by specifying certain parameters through `udev` rules.

Rules explained!

The behaviour of `udev` on handling each of the devices can be controlled by using `udev` rules. Most of the newer distros ship with a number of default `udev` rules meant for hardware detection. When deciding how to name a device and which additional actions to perform, `udev` reads a series of rule files. These files are kept in the `/etc/udev/rules.d` directory, and they all must have the `.rules` suffix. In a rules file, lines starting with `"#"` are treated as comments. Every other non-blank line is a rule and rules cannot span multiple lines. The default rules file can be seen at `/etc/udev/rules.d/50-udev-default.rules`

A rule consists of a combination of matching keys for the device and the action to be done on matching the device. In other words, a rule explains how to find the specific device and what to do when it is found.

The following is the basic syntax of a rule:

```
KEY1="value", KEY3="value", KEY4=="value"...SYMLINK+="link"
```

The following line is a simple `udev` rule. It tells the `udev` daemon to create `/dev/cdrom` and `/dev/cdrom0` softlinks to `/dev/hdc` whenever it finds `/dev/hdc`.

```
KERNEL=="hdc", SYMLINK+="cdrom cdrom0"
```

It is to be remembered that we can specify multiple rules for a single device and it can be written in multiple `.rules` files. When a device is plugged in or unplugged, the `udev` daemon looks through all the `.rules` files in the `/etc/udev/rules.d` directory until all matching rules are read and executed.

The following are some of the keys or parameters that can be used for device matching and the actions in a `udev` rule:

- **BUS:** matches the bus type of the device; examples of this include PCI, USB or SCSI.
- **KERNEL:** matches the name the kernel gives the device.
- **ID:** matches the device number on the bus; for example, the PCI bus ID or the USB device ID.
- **PLACE:** matches the topological position on the bus, such as the physical port a USB device is plugged in to.
- **SYSFS_filename, SYSFS(filename):** allows `udev` to match any `sysfs` device attribute, such as the label, vendor, USB serial number or SCSI UUID. Up to five different `sysfs` files can be checked in a single rule, with all of the values being required in order to match the rule.
- **PROGRAM:** allows `udev` to call an external program and check the result. This key is valid if the program returns successfully. The string returned by the

TABLE 1: OPERATIONS FOR UDEV KEYS

Operator	Meaning
<code>==</code>	For matching. Eg: <code>KERNEL=="ttyUSB0"</code>
<code>=</code>	Setting a parameter. Eg: <code>NAME="my_disk"</code>
<code>+=</code>	Adding to list. Eg: <code>SYMLINK+="cd1 cd2"</code>

program additionally may be matched with the `RESULT` key.

- `ATTR`: different attributes for the device like size, product ID, vendor, etc.
- `RESULT`: matches the returned string of the last `PROGRAM` call. This key may be used in any rule following a `PROGRAM` call.
- `RUN`: it can be set to some external program that can be executed when a device is detected.
- `SYMLINK`: for creating symlinks for the matching device.
- `ACTION`: permits two match conditions ‘add’ and ‘remove’ when a new device is added or removed.

In addition to this, Table 1 lists different operators you can use with each of the keys.

Now the question is: how do we collect information about devices?

Writing a rule is, in turn, matching the device by specifying unique bytes about the device. The unique information about the device can be grabbed from `sysfs`:

```
[slynx@gnubox tmp]$ cat /sys/block/sda/sda1/size
14336000
```

Here I have retrieved an attribute size for the device `sda1`. Now I can use `ATTR{size}=="14336000"` to match the device `/dev/sda1`.

To make the job easier, we have a `udev` utility called `udevinfo`, which can be used to collect details about devices and write rules in a very handy way. The following is the `udevinfo` output for the same `/dev/sda1`:

```
# udevinfo -a -p /sys/block/sda/sda1
looking at device '/devices/pci0000:00/0000:00:1f.2/host0/
target0:0/0:0.0:0/block/sda/sda1':
  KERNEL=="sda1"
  SUBSYSTEM=="block"
  DRIVER==""
  ATTR{dev}=="8:1"
  ATTR{start}=="2048"
  ATTR{size}=="14336000"
  ATTR{stat}==" 190 59 2162 1655 30 31 488 836
  0 1652 2491"
```

As you can see, it returned a lot of information about the device. We will take some of the above lines to make a `udev` rule:

```
KERNEL=="sda1", SUBSYSTEM=="block", ATTR{dev}=="8:1", 2048
```

Now the match is ready! You can even create a symlink for the device as `/dev/musicdrive`:

```
KERNEL=="sda1", SUBSYSTEM=="block", ATTR{dev}=="8:1", 2048",
SYMLINK+="musicdrive"
```

Alternatively, you can use the following to obtain information about any device name:

```
# udevinfo -a -p 'udevinfo -q path -n /dev/devicename'
```

Setting up an automatic Internet connection

I depend on BSNL EVDO/CDMA for Internet access. I have configured the dialling by using the `wvdial` PPP utility, and I issue the `wvdial` command to connect under Fedora 9. I found it interesting to write `udev` rules to auto connect Internet whenever I plug in the EVDO USB modem.

Here’s how to get started: first, plug in the EVDO device in the USB port; second, run the `dmesg` command at a terminal prompt. I received the following `dmesg` output:

```
usb 6-2: New USB device found, idVendor=05c6, idProduct=6000
usb 6-2: New USB device strings: Mfr=1, Product=2, SerialNumber=0
usb 6-2: Product: ZTE CDMA Tech
usb 6-2: Manufacturer: ZTE, Incorporated
```

But there was no suitable kernel module loaded to create `/dev/ttyUSB0`, which is the device node for the corresponding device. You might try manually loading the USB serial module specifying the Product ID and vendor ID parameters—that is, `idVendor=05c6, idProduct=6000`. Run the following command as the root user:

```
/sbin/modprobe usbserial product=0x6000 vendor=0x05c6
```

Executing the `dmesg` command again brings up the following:

```
usb 6-2: configuration #1 chosen from 1 choice
usbserial_generic 6-2:1.0: generic converter detected
usb 6-2: generic converter now attached to ttyUSB0
```

As you can see, this time `/dev/ttyUSB0` is created and made available. [Actually when the module `usbserial` is loaded using the `modprobe` command, it is required to manually create `/dev/ttyUSB0` using the `mknode` command. But there is a default `udev` rule that creates the device.]

Now we have to dial `wvdial` as the root in order to connect. How do we transform this manual process to a `udev` rule? Run the following command to collect appropriate parameters to match the device:

```
udevinfo -a -p $(udevinfo -q path -n /dev/ttyUSB0)
```

Now, create a file called `/etc/udev/rules.d/100-bsnl`.

rules and enter the following rules in it:

```
ATTRS{idVendor}=="05c6", ATTRS{idProduct}=="6000", RUN+="/
sbin/modprobe usbserial product=0x6000 vendor=0x05c6",
SYMLINK+="netdevice"

ACTION=="add", SUBSYSTEM=="tty", KERNEL=="ttyUSBO",
ATTRS{idVendor}=="05c6", ATTRS{idProduct}=="6000", RUN+=""/
usr/bin/evdo_connect

ACTION=="remove", SUBSYSTEMS=="usb", KERNEL=="ttyUSBO", RUN+="/
/usr/bin/msg_connection"
```

The first rule instructs *udevd* to listen to devices with parameters *idVendor*=05c6 and *idProduct*=6000. If found, load the corresponding *usbserial* kernel module. The second rule instructs *udevd* to execute the *evdo_connect* script when the above parameters match for a newly added device */dev/ttyUSBO*. *ACTION*="add" means, when the device was added.

The parameter value for *RUN* is an executable command. But it should be noted that the executable should be something that runs finite times rather than something that contains an infinite loop or infinite conditions.

/usr/bin/evdo_connect is made to run for a finite number of times by sending *wvdial* and *msg_connection* to the background.

Now, create two files. In the first file named */usr/bin/evdo_connect* enter the following text:

```
#!/bin/bash
/usr/bin/wvdial &
/usr/bin/msg_connection con &
```

...and in the second file named */usr/bin/msg_connection*, enter the following:

```
#!/bin/bash
user=slynx ; # Specify the user to which notification is to be shown

if [ $# -eq 0 ];
then
    DISPLAY=:0 su $user -c 'notify-send -u critical "Internet
Disconnected :("';
else
    while true;
    do
        if [[ -n $(/sbin/ifconfig ppp0 2>&1 | grep "inet addr") ]];
        then
            DISPLAY=:0 su $user -c 'notify-send "Connected to Internet :)"';
            exit 0;
        fi
        sleep 1;
    done
fi
```



Figure 1: 'Connected to Internet' notification



Figure 2: 'Internet disconnected' notification

done

fi

In this script, we have used the *notify-send* utility to display messages to the user. *notify-send* comes default with Fedora 9. You may have to install it separately on Ubuntu or other distributions.

Now, set executable permissions to both the scripts since *udev* is going to execute them upon finding the device:

```
# chmod +x /usr/bin/evdo_connect
# chmod +x /usr/bin/msg_connection
```

Voila! The auto dialling is configured and ready to run. As soon as I plug or unplug EVDO now, I get notifications as shown in Figures 1 and 2, in real time.

The procedure is the same while using any other mobile/CDMA Net connection. You have to modify the *udev* rules slightly, according to your device parameters.

Auto syncing a back-up drive

Let's look at a typical problem: I have a back-up hard drive. I used to back up my home directory everyday in this hard disk. This is normally done manually so, again, let's use *udev* to automate the procedure. Again, as we did with the EVDO modem, first plug in the external hard drive. Then

run *dmesg* to identify the device. The following is the *dmesg* output in my case:

```
usb-storage: device scan complete
scsi 7:0:0:0: Direct-Access  HITACHI_DK23DA-20  00J2 PQ: 0 ANSI: 0
sd 7:0:0:0: [sdb] 39070079 512-byte hardware sectors (20004 MB)
sd 7:0:0:0: [sdb] Write Protect is off
sd 7:0:0:0: [sdb] Mode Sense: 03 00 00 00
sd 7:0:0:0: [sdb] Assuming drive cache: write through
sd 7:0:0:0: [sdb] 39070079 512-byte hardware sectors (20004 MB)
sd 7:0:0:0: [sdb] Write Protect is off
```

Now, collect suitable keys to match the device using the following command:

```
# udevinfo -a -p $(udevinfo -q path -n /dev/sdb) | more
```

The output in my case was:

```
looking at device '/devices/pci0000:00/0000:00:1d.7/usb2/2-1/2-1:0'
host7/tar
get7:0/7:0:0:0/block/sdb':
  KERNEL=="sdb"
  SUBSYSTEM=="block"
  DRIVER=""
  ATTR{dev}=="8:16"
  ATTR{range}=="16"
  ATTR{removable}=="0"
  ATTR{size}=="39070079"
  ATTR{capability}=="12"
  ATTR{stat}==" 51 285 456 340 1 0 8
  9 0 278 349"
  looking at parent device '/devices/pci0000:00/0000:00:1d.7/usb2/2-1/2-1:0/ho
host7/target7:0/7:0:0:0/block':
  KERNELS=="block"
  SUBSYSTEMS==""
  DRIVERS=""
  looking at parent device '/devices/pci0000:00/0000:00:1d.7/usb2/2-1/2-1:0/ho
host7/target7:0/7:0:0:0':
  KERNELS=="7:0:0:0"
  SUBSYSTEMS=="scsi"
  DRIVERS=="sd"
  ATTRS{device_blocked}=="0"
  ATTRS{type}=="0"
  ATTRS{scsi_level}=="0"
  ATTRS{vendor}=="HITACHI"
  ATTRS{model}=="DK23DA-20  "
```

Now formulate a matching rule as the following and write to a rule file [we'll call it */etc/udev/rules.d/100-backupdisk.rules*]:

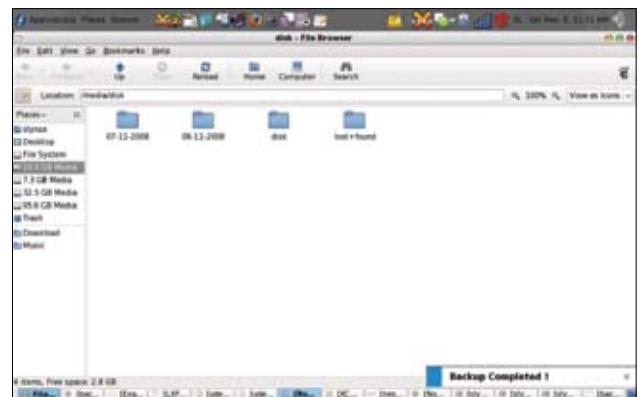


Figure 3: Back-up completed notification

```
SUBSYSTEM=="block", ATTR{removable}=="0", ATTR{size}=="39070079",
SYMLINK+="backupdisk", RUN+="/usr/bin/backup"
```

We have an action script */usr/bin/backup*, which is called when a match is found. Write a bash script with the following contents:

```
#!/bin/bash

backup_dir=/home/slynx # Specify the directory to backup
user=slynx # The user to whom which the message is to be displayed

mount /dev/backupdisk /mnt/backups;

rsync -a $backup_dir /mnt/backups/$(date +%d-%m-%Y) /;

umount /mnt/backups;

DISPLAY=:0 su $user-c 'notify-send "Backup Complete";'
```

Notice that the script mounts the external disk under */mnt/backup*. So, make sure you create that directory as well. Following this, make the script executable as follows:

```
# chmod +x /usr/bin/backup
```

That's it! Now, every time you connect the external disk, it starts the back-up procedure using *rsync* automatically. Once the procedure ends, you will get a pop-up notification on your desktop as well (Figure 3).

You can tweak around a bit to make this back-up drive encrypted as well. However, I'll leave you to try it out yourself.

So, that's all for now. Have fun with *udev*, and happy hacking! 

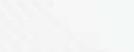
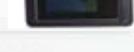
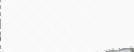
By: Sarath Lakshman is an 18 year old hacker and free software enthusiast from Kerala. He loves working on the GNU/Linux environment and contributes to the PiTiVi video editor project. He is also the developer of SLYNX, a distro for newbies. He is currently studying at Model Engineering College, Cochin. He blogs at www.sarathlakshman.info

Festival Season

Celebration Begins with Offers from

EFYbonanza

Tech Products. Amazing Offers.





Protocols to Transfer Files Between Mobiles and PCs

Find out how to configure the Picture Transfer Protocol and the Media Transfer Protocol on Linux, in order to transfer files between your mobile and PC.

Not so long ago, the 'hand phone' was a pretty expensive device and only the elite could get their hands on one. There were very few service providers and the calling rates were not affordable for everyone. As the market opened up, 'cheap' mobile phones became available even to the common man. Affordability is one of the key factors for the success of the mobile phone market -- it made everybody a gadget freak and turned cell phones into a part of our daily lives.

When consumers looked at a mobile device, they saw much more than *just* a mobile phone. The end users now expected more features in their mobiles and, to take this market to the next level, mobile phone manufacturers started adding more features like a camera, MP3 players, Internet connectivity, and so on. Software standards needed to be established for such extensions so that life for the user became simpler.

Protocols like Picture Transfer Protocol (PTP) and Media Transfer Protocol (MTP) evolved to support gadgets communicating with PCs to transfer images and media files in a standard way. Support for such protocols is inbuilt into the operating system or provided by the gadget manufacturers. Linux requires certain libraries to be installed to support these protocol extensions.

The protocols

Digital Still Photography Devices (DSPD) like digital still cameras are available in plenty from a variety of vendors. Standardisation is required for these devices so that they can interact with the PC or other digital devices like printers of different vendors. Picture Transfer Protocol (PTP) provides a standard way to interact with a DSPD. This protocol provides mechanisms to exchange images to and from the DSPD and PC. It also provides mechanisms to control DSPD and the ability to transfer auxiliary information such as non-image data files. PTP works on top of transport protocols like USB, IrDA, IEEE1394, but is not limited to these.

The Media Transfer Protocol (MTP) is an extension of PTP, promoted by Microsoft to enable media players to effectively and securely manage media files like songs or videos.

Normally, these media devices are exposed as mass storage devices to the PC, and the PC gets exclusive access of the media data. This exclusive access can lead to data corruption. These protocols provide more controlled access. The media files are exposed as files or objects, which are locally maintained by the MTP devices. It also provides secure access to the media files and provides information on the file format and other capabilities.

To enable these protocols in Linux, we need a transport medium. The USB port is one of the most popular transport media for PCs and also in mobile devices. In the following sections we will explore how to bring in a generic USB driver into your kernel and successfully set up these protocol extensions.

Support for PTP and MTP

As we discussed earlier, PTP and MTP sit on top of some transport protocol like the USB. Mobile devices provide the USB interface to connect and sync data with the PC. The first thing we need on our Linux system is a driver for the device. The open source community provides us with a generic USB driver called *libusb*—a USB library that exports APIs to the user space, enabling applications to be developed above it. You can download and install *libusb* from libusb.wiki.sourceforge.net.

A successful installation of *libusb* can be confirmed by running the *testlibusb* command as follows:

```
rajaram@rajaram-laptop:~/libusb-0.1.12/tests$ ./testlibusb
Dev #0: 0000 - 0000
Dev #0: 064E - A103
Dev #0: 0000 - 0000
rajaram@rajaram-laptop:~/libusb-0.1.12/tests$
```

The above output lists all USB devices connected to my PC. Therefore, with the successful installation of *libusb* it now provides us the transport layer support.

To enable picture transfers, PTP is required to be installed. PTP also comes as a library that can be downloaded from libptp.sourceforge.net. A successful compilation and installation can be verified by running the *ptpcam* command. The following snippet shows a successful PTP installation:

```
rajaram@rajaram-laptop:~/libptp2-1.1.10/src$ ./ptpcam --help
USAGE: ptpcam [OPTION]

Options:
--bus=BUS-NUMBER      USB bus number
--dev=DEV-NUMBER       USB assigned device number
-r, --reset             Reset the device
-l, --list-devices     List all PTP devices
-i, --info              Show device info
-o, --list-operations  List supported operations
-p, --list-properties  List all PTP device properties
                        (e.g. focus mode, focus distance, etc.)
-s, --show-property=NUMBER  Display property details (or set its value,
                           if used in conjunction with --val)
--set-property=NUMBER   Set property value (--val required)
```

```
--set=PROP-NAME      Set property by name (abbreviations allowed)
--val=VALUE          Property value (numeric for --set-property and
                     string or numeric for --set)
--show-all-properties Show all properties values
--show-unknown-properties Show unknown properties values
-L, --list-files     List all files
-g, --get-file=HANDLE Get file by given handler
-G, --get-all-files  Get all files
--overwrite          Force file overwrite while saving to disk
-d, --delete-object=HANDLE Delete object (file) by given handle
-D, --delete-all-files Delete all files from camera
-c, --capture         Initiate capture
--nikon-ic, --nic    Initiate Nikon Direct Capture (no download!)
--nikon-dc, --ndc    Initiate Nikon Direct Capture and download
--loop-capture=N    Perform N times capture/get/delete
-f, --force           Talk to non PTP devices
-v, --verbose         Be verbose (print more debug)
-h, --help            Print this help message> 4. Promote LFY through a
FREE booth, and provision of volunteers to man
> the booth and promote the magazine
```

```
rajaram@rajaram-laptop:~/libptp2-1.1.10/src$
```

The next step in the process is to add MTP support to the kernel. Again, the advantage of the open source community can be used here. MTP comes as the *libmtp* library, which could be downloaded from libmtp.sourceforge.net. *libmtp* is a user-space application that uses APIs of the generic *libusb* driver. The list of devices supported can be found from the same site. The following snippet shows a successful installation of MTP:

```
rajaram@rajaram-laptop:~/libmtp-0.3.3/examples$ ./detect
libmtp version: 0.3.3

Listing raw device(s)
No raw devices found.
rajaram@rajaram-laptop:~/libmtp-0.3.3/examples$
```

As an end user, if you look for user-friendly tools with the user interface, there are plenty available on the Internet that use *libptp* and *libmtp* and act as a front-end to the libraries. So, when you plan to buy a mobile next time, don't just look for a mobile -- look for more than *just* a mobile; look for those with PTP and MTP extensions. 

REFERENCES:

- <http://libusb.wiki.sourceforge.net/>
- <http://libptp.sourceforge.net/>
- <http://libmtp.sourceforge.net/>

By: Rajaram Regupathy. The author is a technical lead with HCL and can be reached at rera_raja@yahoo.com

Programming in Python for Friends and Relations: Part 8



Programming in Python for Mobile Gadgets Using the Web

There are two aspects to the Web—one is focused on making information available on the Net and the other on consuming that information. In this article, let us look at the latter.

Everything on the Web is expected to be accessed through the browser. If you are restricted to the screen size of a smart phone, browsing is not much fun. Most of the Web pages are not designed for the small screen. Navigating for what you need is hard. Hence, little applications that you can use to extract and display just what you want, can be very useful. Applications like the stock ticker are available

for many stock exchanges. However, you may not find a little applet for your needs. For example, you may have invested in several schemes from several mutual funds and wish to know the net asset value of each, in a simple table. So, you should be able to develop one.

Probably, the first smart phone to offer Python to develop applications on the smart phone was from the Nokia S60 family, wiki.opensource.nokia.com/projects/Python_for_S60

S60. The www.maemo.org website provides the open source platform and tools for the Nokia 770/800 tablets. Openmoko does not come with the Python interpreter by default, but can be customised to include it (wiki.openmoko.org/wiki/Application_Development_Crash_Course).

Once the interpreter is available with the required modules, you just need to copy your Python source on the phone/device and run it.

Getting started

If you need to extract information from the Web, the first thing your application will need to do is to access a Web page. After a little research, you will find the *urllib2* is the appropriate module. The method you need is *urlopen*. After establishing the connection to a website, you will want to read the page. So, start Python and try the following code:

```
>>> import urllib2
>>> dir(urllib2)
['Iffy', '_urlopen', 'urlopen']
>>> Iffy_home=urllib2.urlopen("http://www.lfymag.com")
>>> dir(Iffy_home)
['Iffy', '_read', '_readline', '_readlines', 'close', 'fileno', 'geturl', 'read', 'readline', 'readlines', 'urlinfo']
>>> Iffy_home.read()
```

The above code is equivalent to looking at the page source after going to "<http://www.lfymag.com>". You need to parse the page source so that you can extract only what you need.

The obvious option is *htmlllib*. But *htmlllib* uses *sgmlplib*. If you are not really interested in formatting the page or following links, then *sgmlplib* is the easier option. It has a test feature also.

So save a page that you are interested in, for example, <http://google.co.in>, one of the simplest pages on the Web. You can get started with understanding the structure and content of the page by trying the following:

```
$ python /usr/lib/python2.5/sgmlplib.py Google.html
```

Replace *lib* by *lib64* on an x86-64 system and the appropriate Python home directory, in case it is not Python 2.5. You will see a lot of output!

Extracting what you want

Your first job is to find the tag and the data in which you are interested. Then find a suitable pattern so that you can select it using a program.

In all probability, you are most likely to want information from a financial or sports site. But let us take a simple example. You love movies and would prefer to decide your evening plans after knowing the films on television. So, you can write an application to extract just the film name and the starting time from the Web page.

Go to the URL of a channel's current schedule, for example, <http://www.utvworldmovies.com/WeeklyListing.php>, and save this page as *WeeklyListing.html*. The local page will help you understand the content and the fields you

need—the name and the time of the show.

Use an html editor, like Quanta Plus or Bluefish, to examine *WeeklyListing.html*. Combine that by looking at the page and the output of the test mode of *sgmlplib* to identify what you need. Usually, the data you are interested in is in an HTML table and *td* tags, possibly enclosed in a *div* tag. In this case, the *div* tag with *id list0* contains the schedule for the current day.

You are now ready to write your code. The nice thing is that all your development can be done on the desktop and then moved to the device. You can do some testing by using the device image and running it on the desktop using Qemu. Write the following code in *film_schedule.py*:

```
from sgmlplib import SGMLParser

class selector(SGMLParser):
    def reset(self):
        SGMLParser.reset(self)
        self.wanted = False

    def start_div(self, attrs):
        if ('id', 'list0') in attrs:
            print "Found the div"
            self.wanted = True

    def end_div(self):
        if self.wanted:
            print 'End of div'
            self.wanted = False

def page_test(html_page):
    f = open(html_page)
    parser = selector()
    parser.feed(f.read())
    parser.close()
```

The SGML parser initially calls the *reset* method. If there is a method *start_tagname*, it will call that method at the start of a tag named *tagname*. The parameters in the tag are passed as a list of name and value pairs. You will need to look at other tags once we are in the desired block. So, use a flag *self.wanted*. Set it to *true* once the desired *div* starts and reset it to *false* once the end of that tag is reached.

While testing, you may feed the parser the saved HTML file. Later, you will call the actual Web page using *urlopen*. Now you can try this code as follows:

```
>>> from film_schedule import *
>>> page_test('WeeklyListing.html')
Found the div
End of div
>>>
```

So, there is only one occurrence of the *div* in which you

are interested. The film name and time are the data in *td* tags with class *listcontent01*. So, you will need to handle *td* tags, but only within the desired *div*. Each row can be identified by the *tr* tag. Further, you will need to capture the data by a method *handle_data*. So, your code in *film_schedule.py* should look like what's shown below:

```
from sgmlib import SGMLParser

class selector(SGMLParser):
    def reset(self):
        SGMLParser.reset(self)
        self.wanted = False
        self.pick_data = False
        self.films = []

    def start_div(self, attrs):
        if ('id', 'list0') in attrs:
            self.wanted = True

    def end_div(self):
        if self.wanted:
            self.wanted = False

    def start_tr(self, attr):
        if self.wanted:
            self.film = []

    def end_tr(self):
        if self.wanted and self.film:
            self.films.append(self.film)

    def start_td(self, attrs):
        if self.wanted:
            if ('class', 'listcontent01') in attrs:
                self.pick_data = True

    def handle_data(self, data):
        if self.pick_data:
            self.film.append(data)
            self.pick_data = False

    def page_test(self, html_page):
        f = open(html_page)
        parser = selector()
        parser.feed(f.read())
        parser.close()
        return parser.films
```

handle_data is a method that we will use to process the data between the tags. Now, run the following code:

```
>>> from film_schedule import *
>>> for film in page_test('WeeklyListing.html'):
...     print film
...
```

```
['8:15 am', 'Three Colours Red']
['10:30 am', 'The Triangle 1']
['12:30 pm', 'The Triangle 2']
['2:15 pm', 'The Triangle 3']
['5:45 pm', 'Sophia Loren\xe2\x80\x99s Birthday: Boccaccio 70']
['8:30 pm', 'Sophia Loren\xe2\x80\x99s Birthday: A Special Day ']
['11:00 pm', 'Liven Up Nights: My Girl']
>>>
```

The desired data is now very compact.

Working with Web data

You will now want to read directly from the Web. So, add the following method in *film_schedule.py*:

```
import urllib2
def get_films(url):
    page = urllib2.urlopen(url)
    parser = selector()
    parser.feed(page.read())
    parser.close()
    return parser.films
```

Now, run the program:

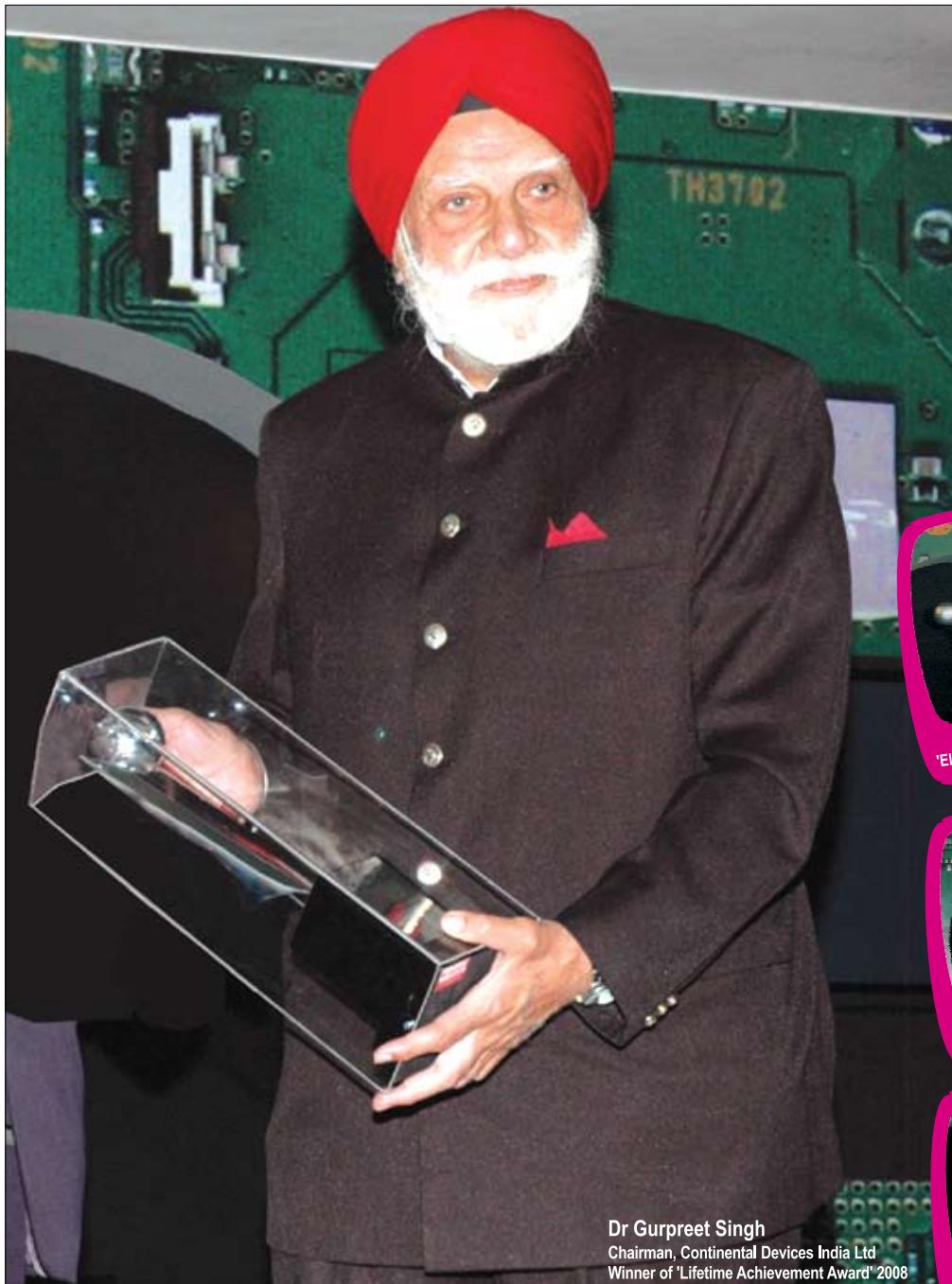
```
>>> from film_schedule import *
>>> for film in get_films('http://www.utvworldmovies.com/WeeklyListing.php'):
...     print film
...
['8:30 am', 'Animation Attack: Rock-A-Doodle']
['10:15 am (World Movies Platinum Collection)', 'World Movies Platinum Collection: Leon']
['12:45 pm', '50 Movies To See Before You Die- Mahesh Bhatt\xe2\x80\x99s Choice: The Great Dictator']
['4:00 pm', 'World Movies for World Peace: The Great Land Of Small']
['6:00 pm', "World Movies for World Peace: Winky's Horse"]
['8:30 pm', 'World Movies for World Peace: Viva Cuba']
['11:00 pm', 'World Movies for World Peace: Iberia']
>>>
```

The results differ because the test file was saved on an earlier day.

Is this a perfect solution? Of course not! The site may change the page logic and your program will stop working. However, a little programming effort is worth it if you are browsing on a small screen with a slow connection. If a site offers an API to access some data, that would be the better option.

You can display the results using the GUI options available on the specific mobile environment. It is important to realise that conceptually, it is no different from programming on the desktop, except that the screen real estate is a serious constraint.





Rajesh Logani, GM, Videocon receiving
'Electronics Man of the Year' award on behalf of
Videocon CMD, V.N. Dhoot



Dinesh Kapur, Country Sales Manager,
APC (UPS Systems)



R.W. Ghai, Director HR, Moser Baer India Ltd.,
(Jury's Special Award)



Subir Chakraborty, Executive VP-Marketing
& Sales (Industrial), Exide, (SMF Batteries)

INDIA'S LEADING TECHNOLOGY AWARDS

Mar 16, 2009

India Habitat Center, New Delhi

SPONSORSHIP OPPORTUNITIES AVAILABLE! Write to info@efyawards.com for more details

Organised By: **EFY**
SINCERELY YOURS

Tech Media Partner: **ElectronicsForYou**

Website Partners: **EFYtimes.com**
IF IT'S TECHNOLOGY IT'S HERE

electronicsForYou.com

EFY Enterprises Pvt Ltd, D-87/1, Okhla Industrial Area, Phase 1, New Delhi 110 020, Ph: 011-26810601-03; Fax: 011-26817563

Let's Visit the 'Libraries'



An introduction to static and shared libraries...

Libraries are an important form of organising, developing, and distributing software. The libraries, especially shared ones, are supported by traditional as well as modern operating systems. Linux supports most of the multi-threading, multimedia and desktop features, commands and utilities through shared libraries (*libpthread.so*, *libGL.so*, *libgtk.so*, *libsed.so* and so on). It also supports a classic framework to install and maintain shared libraries similar to the Windows DLL framework. This article introduces the static libraries and shared libraries from a Linux perspective.

The fundamentals of static and dynamic linking

To start with, let us try out a simple 'hello-world' program:

```
***** hello.c *****/
#include <stdio.h>
int main()
{
    printf("Hello World...!!");
```

```
return 0;
}
```

Now, let us compile the program:

```
[root@localhost nilesh]# gcc -o hello hello.c
```

In this compilation, GCC performs the 'dynamic linking' of 'libc' with the executable 'hello'. This could be observed using the *ldd* command, which displays the libraries on which the executable depends.

```
[root@localhost nilesh]# ldd hello
 libc.so.6 => /lib/tls/libc.so.6 (0xb74a4000)
 /lib/ld-linux.so.2 => /lib/ld-linux.so.2
 (0xb75eb000)
```

To list the unresolved symbols in the executable, we can use the simple *nm* utility.

```
[root@localhost nilesh]# nm -u hello
 w __gmon_start__
 w _Jv_RegisterClasses
 U __libc_start_main@@GLIBC_2.0
 U printf@@GLIBC_2.0
```

So *printf* and *main* (to be specific, *__libc_start_main*) are the two symbols that will get linked dynamically from 'libc'.

Furthermore, let us execute it and see the 'shared library trace' using the *ltrace* utility.

```
[root@localhost nilesh]# ltrace ./hello 2>out
Hello World...!!
[root@localhost nilesh]# cat out
__libc_start_main(0x08048348, 1, 0xbffff1a4, 0x08048370, 0x080483b8
<unfinished ...>
printf("Hello World...!!\n") = 17
+++ exited (status 0) +++
```

This could be avoided simply by static linking, i.e., by specifying *static* in the compilation:

```
[root@localhost nilesh]# gcc -static -o hello-static hello.c
[root@localhost nilesh]# ldd hello-static
not a dynamic executable
```

One more thing worth noting here is the size of the two executables:

```
[root@localhost nilesh]# ls -alh
-rwxr-xr-x 1 root root 4.6K Oct 4 20:52 hello
-rw-r--r-- 1 root root 86 Oct 4 20:52 hello.c
-rwxr-xr-x 1 root root 403K Oct 4 21:15 hello-static
```

The executable 'hello' is almost 100 times smaller than the 'hello-static' executable, which has been produced by static linking. This is one of the evident advantages of dynamic linking.

Static libraries vs shared libraries

Both static and dynamic libraries have their pros and cons.

Static or archived libraries are self-contained, and once linked, the library need not be available at the time of execution. But this comes at the cost of the executable's size and its need to be re-linked for library upgrade.

Shared libraries are those that are loaded by programs when they start. When a shared library is installed properly, multiple applications could use the same, shared library. They also have the following advantages:

- Because of dynamic linking, executables become smaller in size.
- We can override shared libraries when executing a dependant program.
- We can also update libraries and still support programs that want to use older versions of those libraries.

Creating 'hello-world' libraries

To demonstrate the static and dynamic libraries, let us first create the following source files:

```
***** display.c *****
void display()
{
    printf("Hello World ...!!\n");
}
```

...and:

```
***** display.h *****
void display();
```

In the next sections, we will compile *display.c* into a static library and link it with a client application *main.c*. A subsequent section will demonstrate how to compile the same *display.c* into a shared library and link it with a *main.c*.

A 'hello-world' archived library

Creating and linking archived libraries is fairly easy. First, let us write a simple *main* function to invoke the *display()* function from the library.

```
#include <stdio.h>
#include "display.h"

int main()
{
    display();
    return 0;
}
```

Now let us first compile *display.c* to *display.o*:

```
[root@localhost static]# ls
display.c display.h main.c
[root@localhost static]# gcc -c display.c
[root@localhost static]# ls
display.c display.h display.o main.c
```

We will now create a static library using the command *ar*.

Under Linux, the static libraries (alternatively known as 'archived libraries') have the file name format *lib<name>.a*, and we need to specify only *<name>* when we do the linking. Here we will create *libdisplay.a*:

```
[root@localhost static]# ar -rcs libdisplay.a display.o
[root@localhost static]# ls
display.c display.h display.o libdisplay.a main.c
```

The last job is to compile *main.c* to *main.o* and link it with *libdisplay.a*. Note that here, *libdisplay.a* has been linked statically but we have a dependency on *libc* for *printf* and *main*. Also, the *-L* option specifies the place where *libdisplay.a* would be available. One can choose to copy and maintain it at a standard location such as */usr/*

local/lib/ and specify the same with the *-L* option.

```
[root@localhost static]# gcc -o display main.c -L . -ldisplay
[root@localhost static]# ls
display display.c display.h display.o libdisplay.a main.c
[root@localhost static]# ./display
Hello World ...!!
```

A 'hello world' shared library

Before we go for a shared library, let us know more about the Linux loader that is located at */lib/ld-linux.so.2*. This is responsible for performing dynamic linking and loading of the shared objects. This itself is a shared library and we need to link it when we compile *main.c*, which uses the *display()* routine from the shared library.

Under Linux, the shared libraries have the file name format *lib<name>.so* and we need to specify only *<name>* when we perform the linking. Here, we will create *libdisplay.so*.

The process for writing a *main.c* for a shared library is also a little different. Unlike the static linking, we cannot directly call the *display()* function because of dynamic linking.

The following are the APIs that can be used by client applications that have a shared library:

- *dlopen()* loads and opens the specified shared library and returns a handle for further use in the application
- *dlsym()* checks whether the symbol passed in the argument is available
- *dlclose()* to close/unload the library
- *dlerror()* for error handling

It is interesting and informative to go through the manual pages of the aforementioned APIs.

Using these APIs, here we write our *main.c*:

```
#include <stdio.h>
#include <dlfcn.h>

/* Note here. No need to have display.h*/
int main()
{
    void *handle;
    void (*local_display)();
    char *error;

    /* First access the shared library and get the handle .. */
    handle = dlopen ("libdisplay.so", RTLD_LAZY);
    if (!handle) {
        fprintf (stderr, "%s\n", dlerror());
        exit(1);
    }
```

```
/* Search the shared library and get the symbol 'display' .. */
local_display = dlsym(handle, "display");
if ((error = dlerror()) != NULL) {
    fprintf (stderr, "%s\n", error);
    exit(1);
}

/* Invoke the function */
(*local_display)();

/* Close the shared library...*/
dlclose(handle);
return 0;
}
```

Now let's come to the command line and start the compilation:

```
[root@localhost dynamic]# ls
display.c main.c
[root@localhost dynamic]# gcc -c -fPIC display.c
```

Note the directive *-fPIC* passed to the GCC command. This will cause the code to be 'position independent' and loadable anywhere.

Now let us turn it into a shared library called *libdisplay.so*. This could be done using the *ld* command as stated below:

```
[root@localhost dynamic]# ld -shared -o libdisplay.so display.o
[root@localhost dynamic]# ls
display.c display.o libdisplay.so main.c
```

Note the argument *-shared*, which specifies that the library is a shared one.

The next important step is to show the library to the Linux loader. This is achieved by:

```
[root@localhost dynamic]# /sbin/ldconfig -n .
[root@localhost dynamic]# export LD_LIBRARY_PATH=
```

The final step is to compile *main.c* into an executable 'display' and link it together with the *libdisplay.so* shared library.

```
[root@localhost dynamic]# gcc -o display main.c -L . -ldisplay -ldl
[root@localhost dynamic]# ./display
Hello World ...!!
```

It will be interesting to run the *ldd* command on an executable file 'display'.

```
[root@localhost dynamic]# ldd display
libdisplay.so.0 => ./libdisplay.so.0 (0xb75e8000)
libdl.so.2 => /lib/libdl.so.2 (0xb75d6000)
```

```
libc.so.6 => /lib/tls/libc.so.6
(0xb749f000)
/lib/ld-linux.so.2 => /lib/ld-linux.so.2
(0xb75eb000)
```

We can see the dependency of 'display' on *libdisplay.so* being listed here.

Even going further, let us modify *printf* in *display.c* to print "Hello World ...123!!". We just have to recompile the library. There's no need to touch the executable 'display'.

```
[root@localhost dynamic]# vim display.c
[root@localhost dynamic]# gcc -c -fPIC
display.c
[root@localhost dynamic]# ld -shared -o
libdisplay.so display.o
[root@localhost dynamic]# ./display
Hello World ...123!!
```

The role of soft links

The shared libraries are typically maintained at */usr/lib* or */usr/local/lib*. At a time, there can be multiple versions of a given shared library, like *libdisplay.so.0.0*, *libdisplay.so.0.1*, etc. The main executables simply use *libdisplay.so*, which can just be a soft link and made to point to appropriate versions, from time to time.

The library pitfalls

The following are some pitfalls you will encounter while dealing with libraries.

- *C and C++ header files*: If a C++ program is trying to call a 'C library' function, the function needs to be attributed as 'extern C' in the header files. A standard practice for writing a 'C' header file 'header.h' (pertaining to a library of C functions) to cater to both C and C++ applications, is:

```
***** header.h *****/
#ifndef HEADER_H_
#define HEADER_H_

#endif /* __cplusplus
extern "C" {
#endif /* __cplusplus */
```

... header code goes here ...

```
#ifdef __cplusplus
}
#endif /* __cplusplus */
#endif /* HEADER_H_ */
```

- *Sequence and dependency*: Extreme care needs to be taken when static linking the situations where a given executable has dependencies on multiple libraries and they, in turn, have dependencies on each other. In such a case, the base libraries should be linked first, followed by the dependant ones.
- *The loader environment variables*: The loader environment variables such as 'LD_LIBRARY_PATH' or 'LD_PRELOAD' decide the locations and sequence of the shared libraries when they get loaded. With multiple versions of a given shared library, these variables need to carry the correct information so that the appropriate library gets loaded at runtime.

From here onwards

GNU provides 'Libtool' which hides behind all the complexity related to shared libraries and also makes things portable. I'd recommend that you take a look at the 'GNU Libtool' official page at www.gnu.org/software/libtool

Also, the libraries' wiki page at [en.wikipedia.org/wiki/Library_\(computer_science\)](http://en.wikipedia.org/wiki/Library_(computer_science)) has a lot of information. Anyway, the point at the end of the day is to *share it!* 

By: Nilesh Govande. The author is a Linux enthusiast and could be contacted at nileshgovande@yahoo.com. His areas of interest include Linux system software, application development and virtualisation. He is currently working with the LSI Research & Development Centre, Pune.

**FOSTERing Linux is
The Winner of the 'Most Promising
Certified Training Partner'
&
The 1st Institute in India
to produce RHCSs**

**Trying to stretch your
small IT Budget???**



**Lower your costs by taking
advantage of your existing
infrastructure...**

**CUT COSTS
&
Bridge the GAP with
Open Source Solutions**

Get YOUR people Certified on:

- Red Hat Certified Engineer
- RHS333 - Red Hat Enterprise Security: Network Services
- RH423 - Red Hat Enterprise Directory Services & Authentication
- RHS429 - Red Hat Enterprise SELinux Policy Administration
- RH142 Linux Troubleshooting Techniques and Tools
- RH184 Red Hat Enterprise Linux Virtualization
- Clustering with Virtualization
- Mail Server
- Bash Shell Scripting

FOSTERing Linux

Free & Open Source Training Education & Research

SCO- M22, II Floor Old DLF Colony, Sector 14
Gurgaon 122001 Haryana,
Tel: 0124-4268187, 4080880,

55, Deepak Building 13, Nehru Place
New Delhi 110019 Tel: +91-30880047 / 49
Email Id: info@fosteringlinux.com



redhat
TRAINING PARTNER

Total Eclipse: Simplified Java Development with Ingres CAFÉ



Here we look at how to get started with the winner of the LinuxWorld Product Excellence Award for Best Application Development Tool.

The development of Eclipse began in November 2001, and focused on building an open development platform comprising extensible frameworks, tools, and runtimes for building, deploying, and managing software across the lifecycle of a product. Today, Eclipse is one of the most widely used Java development and deployment platforms. However, it can be daunting for developers unfamiliar with such frameworks.

The first challenge developers must face with Eclipse is selecting the various components necessary for their work, such as an application server, Web application framework, database server, and an object-relational mapping layer, to name a few. For the life-cycle management of an application or the use of a development platform by a team of developers, version control and issue tracking modules are also needed. All of this translates into a significant amount of preparatory work that must be completed before starting the Java application development work.

So how can developers get the freedom and flexibility of Eclipse without all that work?

The Ingres CAFÉ Project

Many engineers have considered just that question and a few of them began an open source project to address the issue. Engineering students from Carleton University, the Talent First Network, and Google Summer of Code, along with senior engineers at Ingres Corporation, formed a team to solve the problem. The result is the Ingres Consolidated Application Foundation for Eclipse (CAFÉ), which, in August 2008, won the LinuxWorld Product Excellence Award for Best Application Development Tool.

The CAFÉ team, lead by Samrat Dhillon of Carleton University and Andrew Ross of Ingres, analysed developer requirements to determine the types of components needed by typical Java development teams. The team then researched the Eclipse plug-ins available for each category to determine an optimum platform. Based on this careful research, the team then selected leading components in each category:

- **Apache Tomcat** is the world's leading open source application server. Apache Tomcat makes it easy to test your Java

applications directly from the Eclipse framework. And Tomcat can be used with Ingres CAFÉ in your production environment.

- **Hibernate** is a powerful, high performance object/relational persistence and query service that enables development of persistent classes, including association, inheritance, polymorphism, composition and collections. Additionally, this service allows you to express queries in its own portable SQL extension (HQL), as well as in native SQL, or with an object-oriented Criteria and Example API.
- **Java Server Faces Libraries:** Java Server Faces (JSF) greatly reduces the time required to develop sophisticated, interactive Java applications. Panels and Web pages developed with JSF have rich, sharp graphical controls that are easy to implement. Using JSF also greatly reduces development time, providing significant cost savings. Developers of various skill levels can quickly build Web applications by assembling reusable UI components in a page, connecting these components to an application data source, and wiring client-generated events to server-side event handlers.
- **Ingres Database Community Edition:** Ingres 9.2 is fully integrated and configured with CAFÉ to provide a data repository for application development. Because Ingres 9.2 provides the scalability and security to handle the most demanding business functions, applications developed in CAFÉ can go from proof-of-concept to production, unchanged. Ingres is easy to manage and SQL-92 compliant, so developers familiar with other SQL compliant databases can quickly become productive with Ingres.
- **Ingres Eclipse Data Tools Plug-in (DTP):** The Ingres DTP makes Eclipse Ingres syntax-aware, so developers can quickly and easily develop correct data manipulation statements without Ingres training or experience. The DTP also provides extensive support for data management -- querying of data along with the ability to run SQL statements by simply highlighting the statement text; direct editing of table data to quickly and easily model test data; development of database procedure code; and the ability to execute and get results from row-producing database procedures, etc. All of this makes it extremely easy for developers to become productive immediately with Eclipse and Ingres.
- **Subclipse** is an Eclipse plug-in that provides the functionality to interact with a Subversion server and to manipulate a project in the Eclipse environment. Subversion is a version control system that provides versioning support for directory and file metadata, namespace problems, complexity in administration, and so on. In addition, Subversion provides a versioning control system allowing multiple users to develop on the system at one time. Programmers download the latest version of the code from



Figure 1: CAFÉ installer program

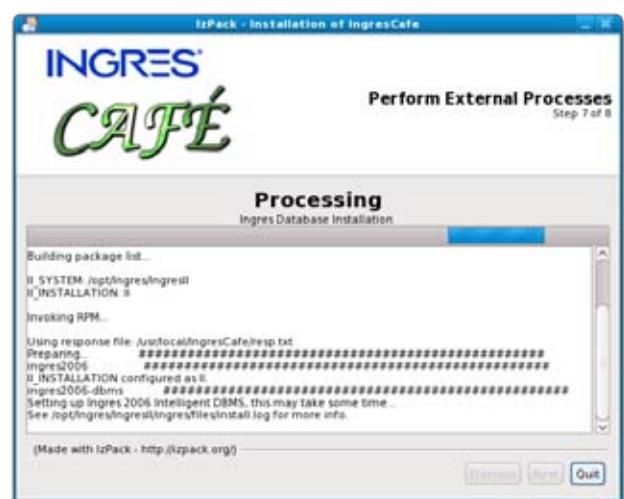


Figure 2: Installation underway

Subversion, make their changes to the code, and then upload the files back to Subversion. Subversion keeps track of the changes and then integrates the changes back into the main code base or notifies the programmer if other modifications were made between their last download and subsequent upload.

- **Mylyn** enhances productivity by seamlessly integrating tasks into Eclipse and automatically managing the context of those tasks as you work. Mylyn extends the Eclipse SDK with sophisticated mechanisms to keep track of tasks. (A task is any unit of work that you want to recall or share with others, such as a user-reported bug or a note to yourself about improving a feature.) Additionally, Mylyn allows you to store tasks locally in your workspace or work with tasks stored in one or more task repositories.

Ingres CAFÉ brings together, in one bundle, all the components developers need to create and deploy rich Java applications, eliminating the time-consuming tasks of acquiring, installing and configuring the many components developers need in a Java development environment.

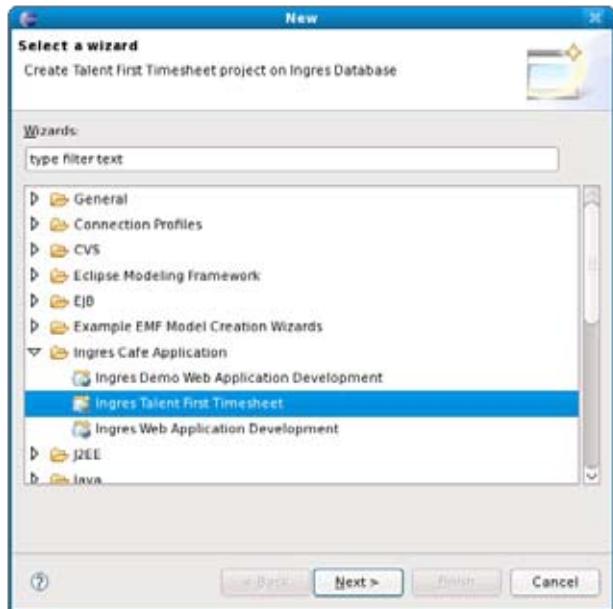


Figure 3: New project window

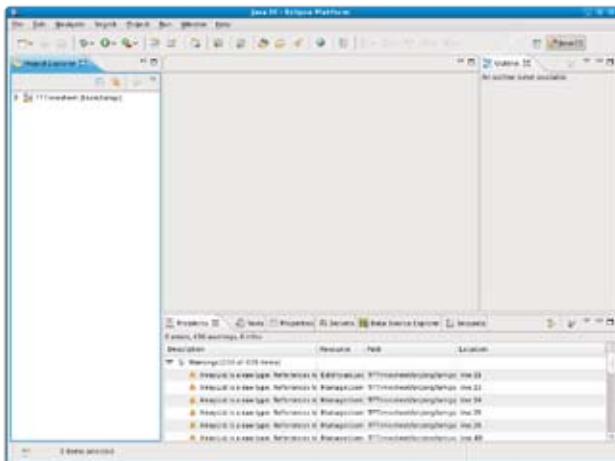


Figure 4: The project explorer

Why Ingres CAFÉ?

Determining the components required for the Eclipse stack is only the first step. Assembling a stack with considerable functionality, by hand, is challenging in terms of the software management, compilation and configuration involved.

Compatibility between software components can often be difficult and time consuming to debug and sort out. This is especially so for people without significant previous expertise, but with a desire to develop Web applications.

CAFÉ lessens the burden of configuration on the user and ensures all components work together naturally. So that anyone—experienced or inexperienced—can simply download and install CAFÉ. It automatically sets up a solid base environment to develop rich Web applications. CAFÉ also provides a special plugin that automatically configures the environment including setting up access to necessary libraries, saving the developer significant time and effort. For any Java developer, whether novice or expert, it's as easy as 'download, install, and start development'.

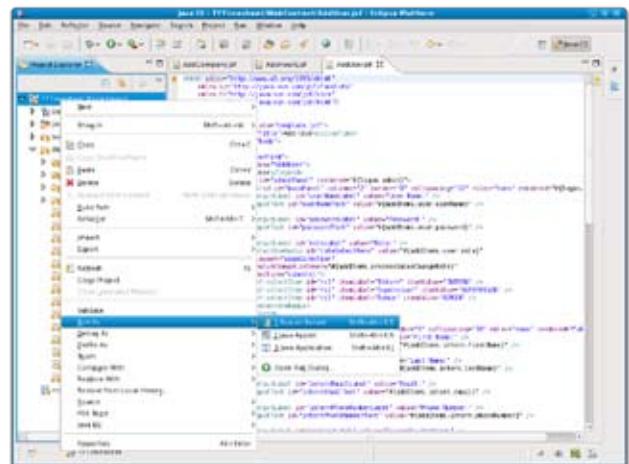


Figure 5: Selecting 'Run on Server' from the context menu

Installing CAFÉ

Before getting started with installation, the most important thing to note, of course, is the system requirements. In order for CAFÉ to run properly, your system must have the following:

- JRE 5.0 or greater
- 512 MB of RAM
- Linux or Windows
- At least 2GB of free disk space

You can install Ingres CAFÉ from the *LFY* CD provided. On Linux, you must log in and install as the root user. You should also make sure that no other packages (for example, updates) are running before attempting to run the install. Copy the Ingres CAFÉ jar file to your hard drive and from a terminal window, type the following:

```
Java -jar CAFE_lin32_rpm_v0.6.0.jar
```

Then press *Enter*. The jar launches the GUI installer. The installer prompts you to accept the licence agreement. Ingres CAFÉ is licensed under the Eclipse Public License—so accept the licence agreement and click *Next*. CAFÉ is installed in */usr/local/IngresCafe* unless you change the path on the next screen. Click *Next* to continue. The following screen allows you to select optional packages to install, so again simply click *Next* to continue. The installation then displays an installation status window. When the packages have been installed, the *Next* button is activated. Click *Next* and CAFÉ configures the various components. When configuration is complete, the installation is finished and you can click *Done*.

Getting started

Start CAFÉ by selecting it from the *Applications* menu under GNOME. Now it's time to get started with Java development. Ingres CAFÉ makes it easy by supplying a working application complete with source code. The Time Sheet Manager is a simple database application that keeps track of the hours people work. It makes use of Ingres' RDBMS (relational

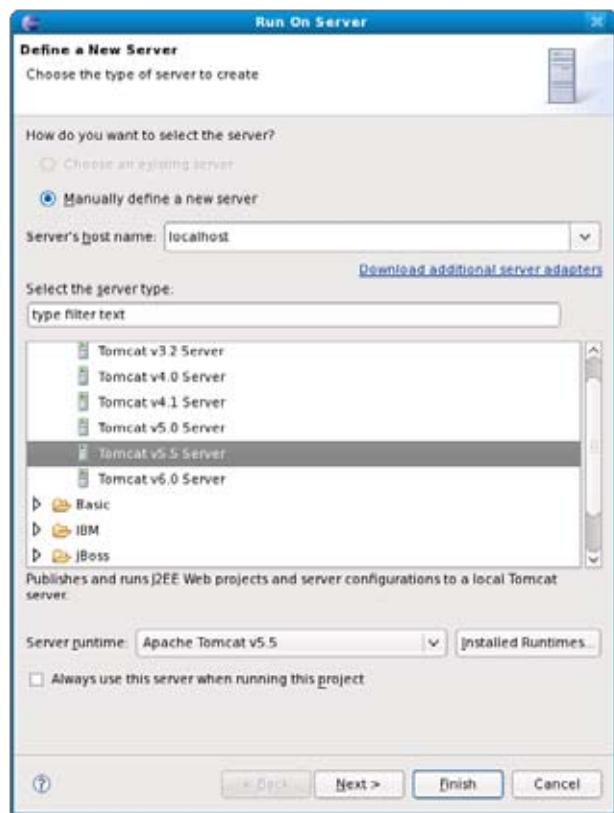


Figure 6: Define a new server

database management system) to create a very simple-to-understand time-keeping system with very little code. The Ingres DTP makes navigating the database schema simple and easy to understand.

You can use the Timesheet Manager application any way you choose. You can use it to keep track of your team's work hours or reuse portions of the application to build new applications. We will now go ahead and create the Timesheet Manager using Ingres CAFÉ!

Creating the Ingres CAFÉ Timesheet Manager

Every application must belong to a 'project' in Eclipse; so first create a project by navigating to *File-->New-->Project*. Choose the Ingres Talent First Timesheet wizard from the list (Figure 3) and click *Next*. The wizard confirms that the '*Timesheet*' project will be created. Click *Finish*.

The project is created and you can see it in the Project Explorer as shown in Figure 4. Congratulations! You have successfully created your first CAFÉ project!

Run the Timesheet

Once the project has been created, you can launch it very easily. Select the project in the Project Explorer and, with the context menu (right click), select *Run As-->Run on Server* (Figure 5).

The run wizard is generic, so it needs to prompt you for a couple of things. First, you choose the application server. Tomcat 5.5 was installed with CAFÉ and should be re-selected

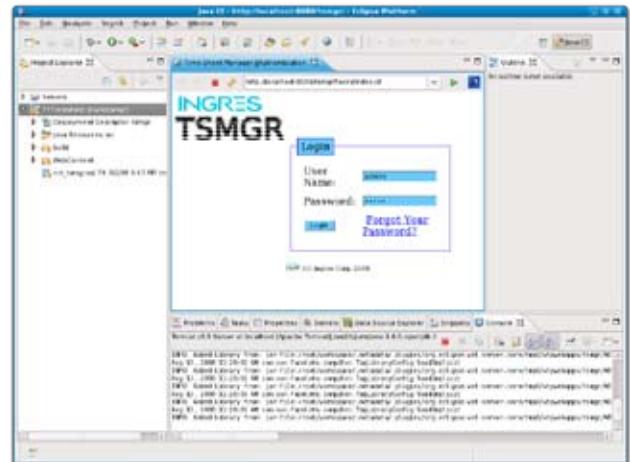


Figure 7: The newly-created application in Eclipse's internal Web browser

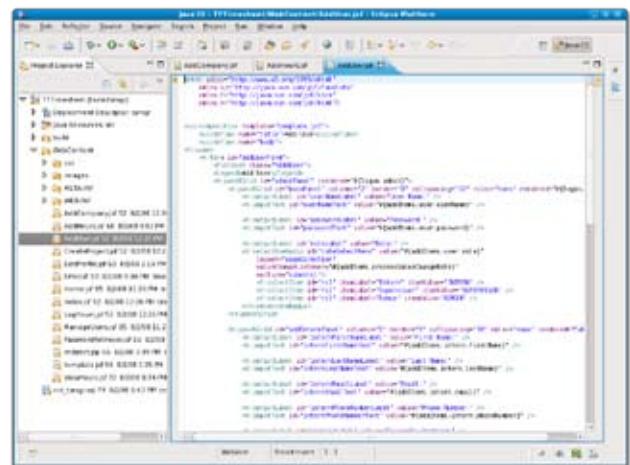


Figure 8: Exploring the Timesheet Manager Java code

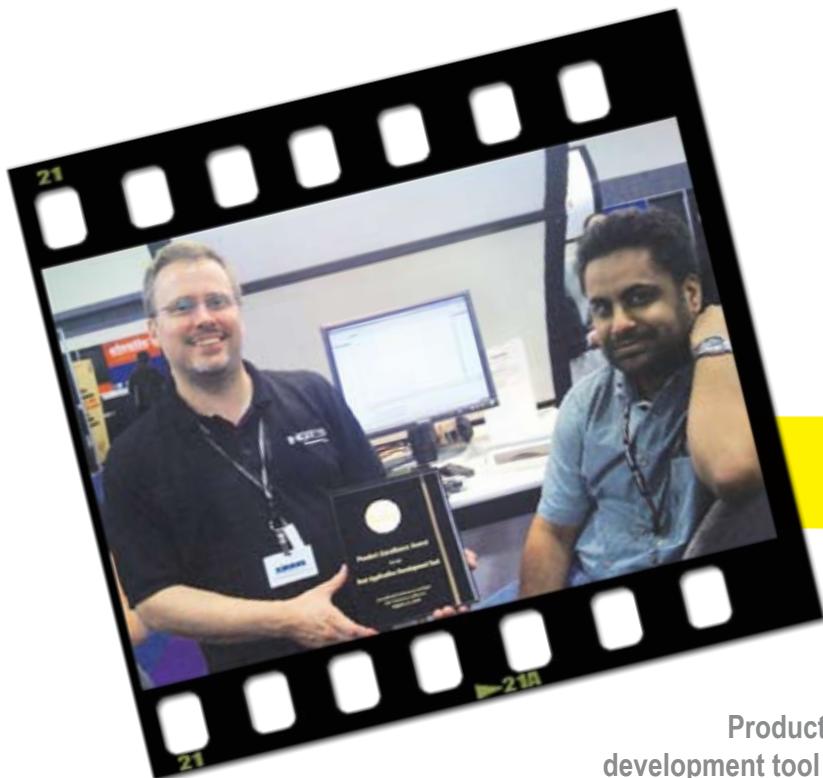
for you. Click *Next* to continue (Figure 6). Second, choose the project just created is pre-selected for you. Click *Next* to continue. The application starts up in Eclipse's internal Web browser (Figure 7).

The Timesheet Manager is a fully functional program that allows you to enter, manage and approve time for any number of workers. It is the program Ingres Corporation uses to manage and track time for interns working on projects. Because this is a complete program, there are many forms and code examples for you to use as building blocks for other applications. You can use the Java perspective to explore the Timesheet Manager Java code (Figure 8) and use that in your own applications.

Now you are ready to go ahead and build your own application using Ingres CAFÉ. Happy programming! 

By: Christine Normile. The author is a senior product manager at Ingres Corporation and has more than 20 years of IT experience in engineering, consulting and marketing in top-tier companies. An accomplished product strategist and marketer, her vision and expertise in relational database management systems have driven notable revenue growth and cost savings for a number of products and companies.

A CAFÉ for Web Developers



The forces behind Ingres CAFÉ, **Andrew Ross and Samrat Dhillon**, with the award.

The Ingres Consolidated Application Foundation for Eclipse (CAFÉ) won the LinuxWorld Product Excellence Award for best application development tool in August 2008. Sam Samrat Dhillon, a master's candidate in the Technology Innovation Management Program at Carleton University, and Andrew Ross, a senior software engineer at Ingres, were the force behind the CAFÉ that was conceptualised keeping the novice users' requirements in mind—for a change. Here we talk to Ross about the nitty gritty of the Ingres CAFÉ.

Q A product is created to address some requirement that doesn't have a solution yet. What was the requirement in your case? Also, was it a personal requirement (as it often happens when you create a product in the OSS domain) or a part of someone else's requirement (e.g., a customer's)?

The key requirement was to make installation and set-up very quick and easy for novice users. This required both research and technical (applied)

problems to be solved.

The first research problem was dealing with overwhelming component choice. There are so many choices of IDE, DBMS, servlet containers/application servers, ORM, and more, that beginners would be overwhelmed before they even started. We wanted to find a good balance of choices that we could make in advance, which would satisfy the needs of novice users.

Samrat's master's thesis, done in parallel to this project, was focused on the distribution of software components with conflicting licences.

Another research problem was to determine the appropriate places to automate configuration and actions on behalf of the developer/user. Then, we applied (wrote and tested) code to do so.

The other key technical problem was to deliver this software with an installer that would provide a positive experience. We aimed for installation in five clicks or less and a time lapse of 15 minutes maximum (for a typical modern computer).

Q. Whose idea was it to come up with a development stack targeted to novice users?

Samrat's research interest is in Web stacks, so he came up with the idea to create a Web stack. There was influence from his thesis co-supervisor, Dr Tony Bailetti, who is very active in the open source community. I also got involved with the project fairly early, and it was my idea to target novice users and how to go about doing so.

Since winning the LinuxWorld award for the best application development tool in August, we have seen talented new developers join and contribute significantly. They have done much of the work for the latest release and have been doing a great job maintaining the project. They work out of Germany.

Q. What prompted you to choose Eclipse as the IDE? Why not NetBeans?

A number of key people in the Eclipse community are Carleton University alumni. This lent itself to the project as we had good access to them and to businesses developing on top of Eclipse. We tried out NetBeans and there were definitely many things we liked about it. In the end, we had to make a tough decision between great IDEs, which is a good problem to have. The Eclipse community, based on what we saw, was larger and had very broad support from industry.

In general, the choice between great technologies was a constant theme in the project. Rather than try to be all things to all people (and increase the risk of failure), we chose to specialise and target a niche.

Q. Would I be correct in saying that CAFÉ, as the name suggests, is simply a consolidation of applications, and thus simply a bigger package of smaller components that are already available?

This is correct. The heart of CAFÉ is a bundling of components that are already available on their own. In addition, there is some glueware and installer code that enhances the overall experience.

Q. You earlier mentioned Samrat's master's thesis that was focused on the distribution of software components with conflicting licences.

Can you elaborate on how licensing issues were taken care of since many of the individual components are released under different licences, which are not necessarily compatible, and then the aggregate, that is CAFÉ, was released under EPL?

Samrat's thesis provides a more elaborate mechanism for distributing conflicting software components. For the sake of brevity, I will not go into details here.

In the case of CAFÉ, the stack includes EPL, Apache, GPLv2, and other licences. Of course, GPLv2 is often recognised as being in conflict with these licences, as clauses in both licences cannot be true at the same time. Since the components are not derivative works of the GPLv2 components, we avoided 'contamination' of the overall stack. The CAFÉ plug-in itself and the overall stack is under EPL. This approach is much like Linux distributions. The demo applications we provided do depend on Ingres (GPLv2 license) and thus it made sense to distribute them as GPLv2.

Q. What's the difference if one chooses to install the components separately? Wouldn't that give a developer greater flexibility?

It is important to remember that we intentionally decided to satisfy the novice user. We observed that there are many power users/developers that roll their own stacks today. Thus, we felt their needs are generally met.

The unaddressed need was to provide an environment for people with less experience. An example to illustrate this point comes from the early days of Linux. There were users who chose to compile their own distribution source code from scratch. This offered incredible flexibility and control. It was also very time-consuming and had a critical prerequisite of the knowledge and skills to do so. The popularity of Ubuntu, Fedora and other distributions is testament to the fact that most people do not want to bother with rolling their own—they want the software to just work. Thus it depends on who you are trying to serve. We want to help the novice user.

The bottom line is that to download, install and configure the components provided with CAFÉ will take you more time and effort than simply installing CAFÉ. This question also assumes you know which component line-up you are going to use in the first place. If you need to research and decide between multiple components for each constituent of the stack, you could be looking at weeks of effort.

Q. CAFÉ is targeted as a platform for Java Web application development. How, and on what basis, did you narrow down on the components that are part of the final CAFÉ stack? I mean, some of the components have a lot of competing alternatives; so how did you finalise your selection?

This was the most challenging and time-consuming aspect to the project. We had to try a number of choices and decide on what grounds we would select the components. We evaluated based on technology strength, community strength, and overall fit with the stack we were creating.

There will be people who prefer NetBeans to Eclipse, Spring to JSF, JBoss to JSF, and so on. We believe that in these cases people are likely to stay with their preference yet respect the merits of the selection we made. Perhaps people will find the most interesting choice to be Ingres, given the popularity of MySQL. This is worth focusing on.

The other components, such as Eclipse, Tomcat, and even the plug-ins we've included, like Subclipse and Mylyn, allow the CAFÉ framework to grow with the developer/user as their needs grow. For the DBMS, there were differences worth factoring. These differences manifest themselves in scalability for transactional processing, robustness, and availability. Not surprisingly, our connection with Ingres gave us an awareness of the differences between MySQL and Ingres. For these reasons, we felt Ingres was a good choice to start (and largely transparent to the novice user). As the application becomes essential to business, not having to rip and replace the DBMS is a huge benefit.

Q. However, what if a developer has a preference for a certain other component—e.g., MySQL (or PostgreSQL) for the RDBMS, or some other app server instead of Tomcat? How do you plan to address this need?

Great question! The interesting thing about CAFÉ is that we did not take steps to prevent power users from adding, removing or swapping components. That capability is still there.

In our humble opinion, in some cases, offering infinite choice and flexibility is where open source can sometimes suffer when compared to closed source competition. Please do not misunderstand us -- we feel choice is very important. We also feel that most people want an option that just works out of the box.

Our thinking is that the typical user we were trying to help would be well served by our choices. Those that want the software to just work should be pleased. Those that want to tinker could still do so.

We recognise that some will be offended by the notion that we chose for them and conversely, many will be appreciative when the software just works. In the end, we never lost focus that it is the latter we were trying to help. We are counting on the community demand to pull us in the direction they want to go in this regard.

Q. Now, many of these components that comprise the stack are also available from the software repositories of distros (OS) the

developer is using. How do you take care of version conflicts or duplicate installations?

Another great question! We recognised this issue in the early stages of the project. To solve it, after considering other options, we chose simplicity, which meant installing CAFÉ into a dedicated directory tree. Redundant software/duplicate installations seemed the lesser of evils. Not surprisingly, the power users in the community shared their strong opinion that we should use components that are already installed, if they are present. This was not practical without significant effort at the time.

When we started the project, we wanted to enable reuse of software already present or available. Thus, CAFÉ would be a logical bundle based on prerequisites. Achieving the ease of use and experience we wanted to provide was not practical based on the software management technology available at the time. This is an opportune time to insert a plug for the Eclipse p2 project. p2 is the next generation plug-in/provisioning system for Eclipse. It is aimed at solving these types of issues with a consistent look and feel across platforms. We intend to make good use of it to enable the kind of software reuse desired with the user experience we want to provide.

Q. Do you plan to talk to distro vendors like RH, Novell and Mandriva to include Ingres Cafe as part of this distribution?

A way to make it much easier for people to start developing is to provide them the software out of the box with the operating system. We are pleased with the success of the versions of CAFÉ so far. We are very grateful for the interest. We believe there is still work to be done for portability and to make CAFÉ even easier to use. When the time is right, we would like to help the distributions make CAFÉ available.

Andrew, thanks for taking time to answer our questions.

Samrat now has successfully defended his thesis (congratulations!) and has accepted work with an IT and professional services firm. Andrew is still involved with the project, although he has been occupied with two new initiatives. The first is to develop technology to store map data in the Ingres relational database, and the second relates to powerful routing and geocoding software. This project involves people in Japan, China, Spain, India, the United States, and other countries. The second is Open Source Bootcamp [osbootcamp.org], a mini-conference devoted to skills development with open source driven by industry, academia and the community. According to Andrew, Open Source Bootcamp held 13 events in 2008 and plans to expand in 2009. 

By: Atanu Datta, LFY bureau

Which one is your career **goal?**



medical



engineering



computer software



computer hardware



hotel management



clinical research

We help you **decide**

Hundreds of Educational Institutions Offering
Professional Courses & Training Under A Single Roof

EduTech's Highlights :

- * An event totally focused on professional education
- * A one-stop facility for students seeking admission into professional courses
- * Counselling by industry top-shots (HR and recruitments)
- * Major plan for advertising and promotion of EduTech Expo 2009
- * Powered by EFY Group's vision to boost professional education
- * Courses catering school pass-outs & fresh graduates

INDUSTRY NEWS



Bilski ruling to end software patents?

In April this year, the Free Software Foundation (FSF), through its End Software Patents (ESP) campaign, filed an amicus brief endsoftpatents.org/bilski to the US Court of Appeals for the Federal Circuit (CAFC), in their en banc hearing of *in re Bilski*. The FSF described the hearing as “an historic opportunity to fix the US patent system, as the Bilski rehearing will directly address the boundaries of the subject matter of patents.” On October 30, 2008, the CAFC issued its ruling, and in it the ESP campaign sees a victory on the path to ending software patents.

As opinions form about the extent to which the Court ruling impacts the patenting of software, one thing is clear—the State Street ruling that in 1998 opened the flood gates to the patenting of business methods and software, has been gutted, if not technically overturned. The vast bulk of software patents that have been used to threaten developers writing code for a GNU/Linux distribution running on general-purpose computers have, in theory, been swept away. The State Street ruling said that you could patent an item if there was a “useful, concrete and tangible result”. In the Bilski ruling, the CAFC have set aside State Street and left us with what they believe to be a simplified test for patentability -- the machine or transformation of matter test: “Thus, the proper inquiry under section 101 is not whether the process claim recites sufficient ‘physical steps’, but rather whether the claim meets the machine-or-transformation test. As a result, even a claim that recites ‘physical steps’ but neither recites a particular machine or apparatus, nor transforms any article into a different state or thing, is not drawn to patent-eligible subject matter. Conversely, a claim that purportedly lacks any ‘physical steps’ but is still tied to a machine or achieves an eligible transformation passes muster under section 101.”

Does the process of loading software on a general-purpose computer become a “particular machine” eligible for patenting? As Professor Duffy of Patently-O recently noted, the Patent and Trademark Office Board of Patent Appeals in two recent non-binding rulings (*Ex parte Langemyr* and *Ex parte Wasynczuk*) outlined its position on the matter: “A general-purpose computer is not a particular machine, and thus innovative software processes are unpatentable if they are tied only to a general-purpose computer.”



The Bilski ruling undoubtedly represents a breakthrough for free software and a success for the FSF’s campaign. But already software patent attorneys are formulating new incantations that they hope will fool the patent examiners into granting software claims, and are instructing their clients to reissue patent applications for pre-existing claims based upon their new theories. Lobbyists for the tech industry are talking of new legislation, and the Federal Trade Commission has announced hearings beginning in December to address recent changes in the patent system: www.ftc.gov/opa/2008/11/ipmarketplace.shtm.

Patent troll attacks Openmoko

“We are sorry that currently we have to remove all the images on the download server of Openmoko. [http://downloads.openmoko.org/release/...](http://downloads.openmoko.org/release/) We will make another stable release as soon as possible. In the mean time, we could rebuild those old releases without mp2/mp3...” The message was posted by Openmoko systems admin Ray Chao in the project mailing list on November 12 under the subject line of “IMAGE/MP3 licensing issue...”

A sort of scary subject line, but the e-mail has no details on it. When someone asked for more details, Wolfgang Spraul responded: “The short story is that we are in a protracted battle with some patent trolls. Google for Sisvel. In order to get ourselves in a stronger position, we want to make sure no copies/instances/whatever of patent-infested technologies like MP2 and MP3 exist on our servers. Our phones never shipped with end-user MP3 playback features, but we want to use this opportunity to make sure it’s not even in some remote place somewhere. For us, the important thing is to defend the freedom of our users rather than cripple our phones so that certain things become ‘impossible’. So please bear with us, while we go through this house cleaning effort.”

So, apart from the house-cleaning effort, what else are the folks at Openmoko up to in response to the attack? “We looked at several options, OIN, patent-commons, peer-to-patent... In the end, we decided to collaborate with the Software Freedom Law Centre in New York. We believe this is most in line with the goals of the Openmoko project, and will have the best long-term results. I cannot speak about details yet; the SFLC and Sean [Moss-Pultz, Openmoko CEO] are working on this. I think next year, with regard to patents, the results from that will be one of the more important developments for Openmoko and maybe even the larger Free Software scene,” wrote Wolfgang.

INDUSTRY NEWS

Szulik is Ernst & Young's Entrepreneur of the Year

Red Hat's chairman, Matthew Szulik, has been awarded the US national winner for the Ernst & Young Entrepreneur of the Year 2008. In addition, Szulik was also named winner in the technology category. Szulik will now represent the US at the *World Entrepreneur of the Year* awards in Monte Carlo, Monaco, where winners from more than 40 countries worldwide will participate.



Szulik has served as chairman of Red Hat's board of directors since 2002 and also as the former president and CEO of the company until late 2007. He has led early-stage technology companies for more than 20 years and joined Red Hat in 1998 as president. He shared Red Hat founder, Bob Young's belief that the collaborative approach of open source and a great brand could redistribute the economics of the technology industry from vendor to customer.

The Ernst & Young Entrepreneur of the Year programme, now in its 22nd year, recognises men and women from around the world who excel at growing and sustaining industry-leading businesses. Past winners include entrepreneurs behind recognisable brands such as Amazon.com, America Online, eBay, Starbucks Corp and Under Armour.

RH authorised repurchase of its common stock

Red Hat has announced that its board of directors has amended the company's previously announced program for the repurchase of its common stock. Now the company is authorised to purchase up to an aggregate of \$250 million of the company's common stock, without regard to amounts previously repurchased under prior programs.

"With today's [November 18, 2008] announcement, we are increasing our capacity to repurchase Red Hat stock," stated Charlie Peters, executive vice president and CFO of Red Hat. "We believe that repurchase programs enhance shareholder value and demonstrate our confidence in the strength of Red Hat and its long-term opportunities."

The amended program will expire on either (i) October 31, 2010, or (ii) a determination by the company's board of directors, CEO or CFO to discontinue the program (whichever occurs earlier). Repurchases of common stock may be effected, from time to time, either on the open market or in privately negotiated transactions. Red Hat had approximately 190.1 million shares of common stock outstanding as of November 13, 2008.



Nokia, Oulu support realXtend project

Adoption of the realXtend open source virtual reality platform is accelerating, extending the support of an interconnected network of 3D virtual worlds with multi-user experiences. Nokia and the City of Oulu, Finland have joined the supporters of the realXtend project aimed at developing the world's best virtual world platform on an open source basis. The key developers are LudoCraft Ltd, a games studio, and Admino Technologies Ltd, specialists in scalable server technologies.

GNU FDL is now v1.3

The Free Software Foundation (FSF) has announced the release of version 1.3 of the GNU Free Documentation License (FDL). This version of the licence allows public wikis to relicense their FDL-covered material under the Creative Commons Attribution-ShareAlike (CC-BY-SA) 3.0 licence.

This new permission has been added at the request of the Wikimedia Foundation, which oversees the Wikipedia project. The same terms are available to any public wiki that uses materials



available under the new licence. The Wikimedia Foundation will now initiate a process of community discussion and will vote to determine whether or not to use CC-BY-SA 3.0 as the licence for Wikipedia.

"Wikis often import material from a wide variety of sources, many of which use the CC-BY-SA licence," said Brett Smith, licensing compliance engineer at the FSF. "Wikipedia, however, uses the GNU FDL. The incompatibility between these two licences has been an obstacle to moving material back and forth between these sites. The new provision of FDL version 1.3 will give Wikipedia and other wikis another chance to choose the licensing policies they prefer."

Version 1.3 of the GNU FDL also adopts the licence proxy and termination clauses that are part of the GNU General Public License version 3, released last year. The full text of the new licence, along with more information, is available at www.gnu.org/licenses/fdl-1.3.html. The text of CC-BY-SA 3.0 is available at creativecommons.org/licenses/by-sa/3.0/.

INDUSTRY NEWS



Drupal wins best overall 2008 open source CMS award

Packt's annual Open Source Content Management System (CMS) Award has announced Drupal as the overall winner, collecting a first prize of \$5,000. Three months after it was launched and a staggering 20,000 votes later, Drupal finished ahead of Joomla! and DotNetNuke to retain the Award it won in 2007.

"These awards are a testament to the valuable contributions from dedicated Drupal community members around the globe," said Buytaert in response to the news. "Working together, the Drupal community is building the future of the dynamic Web so that anyone can quickly build great social publishing websites."

Finishing in second place and receiving \$3,000 was Joomla, the youngest of the three finalists and a previous winner. In third place and receiving \$2,000 was DotNetNuke, the only CMS in the final that is written in VB.NET for the ASP.NET framework.



Ingres, RH offer enterprise-class FOSS platform to ISVs

Ingres and Red Hat have announced plans to work closely together with ISVs in the EMEA region. They are collaborating to present ISVs with an enterprise-class business offering. Also on the agenda is the launch of the Ingres and Red Hat integrated technology platform. Red Hat and Ingres are combining their enterprise solutions in order to provide partners with a powerful solution stack through the combination of three core components -- the Red Hat Enterprise Linux 5 operating system, Ingres Database and JBoss Enterprise Middleware.

Google and Motorola join GNOME Foundation

The GNOME Foundation has announced that Motorola and Google have joined its advisory board. With this, the GNOME Foundation continues to strengthen its industry support and shows that the support for free and open source software is growing, especially in the mobile space with technologies like GNOME Mobile. The additional funds and resources will be used on programmes that support GNOME's goal of universal access such as accessibility outreach programmes, usability studies and internationalisation efforts. GNOME is building on its strength of an accessible desktop to enable universal access to technology through desktops, netbooks and mobile devices.

The Foundation is a non-profit organisation committed to supporting the advancement of GNOME. It provides financial, organisational and legal support to the GNOME project and helps determine its vision and roadmap.

Sun cuts 6,000 jobs, plans restructuring

To align its cost model with the global economic climate, Sun Microsystems is planning to cut 6,000 jobs, or 18 per cent of its global workforce. The company's board of directors has approved the restructuring plan aimed at reducing costs by approximately \$700 to \$800 million annually. Sun expects to incur total charges in the range of \$500 to \$600 million over the next 12 months in connection with the plan, of which it expects to incur approximately \$375 to \$450 million within its current fiscal year 2009.

"Today, we have taken decisive action to align Sun's business with global economic realities and accelerate our delivery of key open source platform innovations—from MySQL to Sun's latest Open Storage offerings," said Jonathan Schwartz, chief executive officer, Sun Microsystems.

Anil Gadre has been appointed as executive vice president of the newly formed Application Platform Software group. Gadre will move from his position as chief marketing officer to lead this new group. The unit, according to the company, will build on its open source leadership position to capitalise on the global market's demand for open application platforms for everything from databases to business integration services on servers, desktops and handheld devices. This includes the entirety of Sun's Java technology franchise, MySQL open source database products, as well as software infrastructure, including the GlassFish Application Server and identity management products. This group will also include the Sun Learning Services organisation.



Intel joins Taiwan to set up Moblin lab

Intel has signed an agreement with the Taiwan Ministry of Economic Affairs (MOEA) to jointly establish an enabling centre for Moblin open source software and applications optimised for Intel Atom processor-based devices.

Paul Otellini, president and CEO, Intel, also announced that, subject to closing conditions, Intel's global investment organisation, Intel Capital, intends to invest \$11.5 million in Taiwanese carrier VMAX. Intel Capital's intended investment and Intel's accompanying business engagement will enable VMAX to deploy Taiwan's first mobile WiMAX network, which is to be commercially available within the first half of next year.



Internationalisation and Localisation: The Tasks Ahead

Resources for localisation need to be set up, and translated too. True, it is not very challenging work, but it needs to be done—maybe as student projects?

Internationalisation (i18n) and localisation (l10n) are two sides of the same coin that deals with software in a multi-cultural world. There are three aspects to all application software: the source code, the presentation layer that the user sees, and the content that is either added by the developer or by the user. Again, the content is divided into static content that remains the same

through the lifetime of the application, or is rarely changed, and dynamic content that changes rapidly.

If you take applications like social networking sites and content management systems, the dynamic content is usually stored in a database as a series of paragraphs, strings, photos, etc, and the pages are generated on-the-fly, depending on the *http* request.

In a multi-cultural, multi-lingual world, all this has to be delivered in the local language and conform to local culture and traditions. i18n and l10n address these issues.

A note on Unicode: There are a large number of languages in the world and a large number of scripts to write these languages. Traditionally, computers only understood ASCII and 7-bit encoding—128 characters. This meant just the English alphabet, numerals and signs. This later became 8-bit encoding, embracing 128 characters. Now, with Unicode, it is possible to encode all the world's languages. Covering that aspect in detail is beyond the scope of this article, but most programming languages have excellent Unicode implementation nowadays, so it is just a question of getting used to using it.

Internationalisation (i18n)

i18n deals with preparing a specific application for translation into any language. It mainly deals with language, although things like the date format, currency and the like are also touched upon. Note that source code is always written in English and does not fall within the scope of i18n, as the end user does not get to see this.

Static strings

To internationalise an application, the first step is to mark the strings that are to be translated. This is done in the source code. Here's a simple example if you have code like this:

```
print 'Hello world'
```

This code will print "Hello world" in English. So, in Python, you would use the following command:

```
print ugettext('Hello world')
```

...where *ugettext* is the Unicode version of the *gettext* module. This is a bit laborious, so the command we use is:

```
import ugettext as _
```

...and mark the string as shown below:

```
print _('Hello world')
```

When all the strings are marked, one can compile the marked strings into a 'pot' file. To create the *.pot* file, you can use the *xgettext* utility. A detailed discussion of this is out of place here, but most languages and frameworks have wrappers around this tool—for example, the Django Web framework has a utility called *makemessages.py*, which when run from the root directory of the application, creates a *.pot* file containing all the strings in the application.

Here is a part of a typical pot file:

```
"Project-Id-Version: Sponsorship system\n"
"Report-Msgid-Bugs-To: \n"
"POT-Creation-Date: 2008-04-26 09:40+0530\n"
"PO-Revision-Date: 2008-04-26 16:27+0530\n"
"Last-Translator: xxxxxxx <x@foo.com>\n"
"Language-Team: xxx <x@foo.com>\n"
" MIME-Version: 1.0\n"
"Content-Type: text/plain; charset=UTF-8\n"
"Content-Transfer-Encoding: 8bit\n"
"X-Poedit-Language: English\n"
"X-Poedit-Country: INDIA\n"
"X-Poedit-SourceCharset: utf-8\n"
```

```
#: templates/base.html:24
#: templates/basext.html:22
msgid "login"
msgstr "kirjaudu"
```

As you can see, there is a msgid "login" and a msgstr "kirjaudu", which is the Finnish translation of the word 'login'. So a translator just takes the English 'pot' file and translates the msgids into his language. Then the translated file is placed under the root directory of the application—typically, under a directory called 'locale'. So for Finnish (language code 'fi'), this would be: *~/locale/fi/LC_MESSAGES/filename.po*. This file is then compiled into a *.mo* file. The compilation is done by a tool called *msgfmt*. When called with the *-a* option, this tool will visit all the language directories and compile the *.po* files it finds there to *.mo* files. Here again, most languages have a wrapper around this to make things easier.

When the application comes across any marked string it has to render, it will check which language it is supposed to use, check if such a language has a *.mo* file; and if there is a file, it will render the string in the language if a translation is available. Otherwise, it will render it in English. Simple! The only problem here is to make sure that this marking of strings is done while writing the code itself. It is a pain to try and mark the strings after the code is written. Fortunately, nowadays all good programmers make sure their code is i18n compliant from the very outset.

The example I've used is in Python—other languages have their own way of marking strings, but the format of the *.pot* file is standard. So translators do not have to worry about what language the application is in—they just translate the strings and hand them over. It is important to note that the potential translator need not have any knowledge of programming or of the source code. He need not even know what the application does. He only needs to know the source and target languages.

There is an excellent tool called KBabel [*kbabel.kde.org*] that automates a lot of the translation, as it can build a database of standard strings in a language that makes it much easier to translate. Marking currency, the date format, number format and the like is done in a similar manner.

Static content

Translation of static content is done the hard way—one file for each language, and depending on the browser/user request, the appropriate language file is chosen for display.

Dynamic content

There are many methods to translate dynamic content. Dynamic content is usually stored in RDBMS and hence is easy to manipulate. I am developing an application that does it as follows: A Web page is made up of building blocks—strings, paragraphs, titles, subtitles—all of which need to be translated and things like images that need not be translated. Let's take a paragraph as an example. In my database, I would have the following tables:

1. Page table, which would consist of a unique title and an ID.
2. One or more paragraph tables that would take a foreign key to the page table and contain a text field for the paragraph. Also, a position field for the position of the paragraph.
3. Then, there is a language table with the codes of the languages I intend to support.
4. Finally, a translation table that will have foreign keys to both the paragraph table and the language table, and will contain the translation of the paragraph in the appropriate language.

If people want to translate, they select a language, the application presents each untranslated string to them, and they translate it.

When a page request comes in, the application checks the desired language and retrieves the translated string/para if available; otherwise, it renders the English equivalent.

This is still work in progress, but anyone interested may see the code at registration.fossconf.in/code/browser/branches/quadmulf, and anyone who wants to contribute is most welcome to!

Where do we stand?

The vast majority of applications today are internationalised—the need of the hour is to provide translations in Indian languages. Except for some major applications, very little work is being done in this field. I don't know whether it is because people are not aware of the need, are too lazy or they do not know how to! What I do know is that not enough is being done in these areas, which takes us to the following section.

Localisation (l10n)

Where i18n deals with making a specific application usable worldwide, l10n deals with making all applications usable for a particular region or locality—often referred to as a locale. This is a very deserted field in our country and even less work is being done here than in i18n. l10n work basically means building accessible databases, usually in

XML format, which all applications in a particular locale can use. Here are some of the pending tasks:

1. We need a database that has the states in India; clicking on a state should give a list of districts in the state; clicking on a district should give a list of towns; each town should lead to a list of localities/mohallas, etc—I am sure you get the idea.
2. Parliamentary, assembly constituencies, including reserved seats.
3. A pin code directory and reverse pin code directory.
4. Telephone area codes.
5. To fill in forms, we require lists of religions/sects/subsects—for example, in England, Christians would be divided into Catholics, Anglicans, Lutherans, etc. In India, we would have Roman Catholics, Syrian Christians, CSI, CNI, Orthodox, etc. Similarly, for other religions, castes and communities.
6. For each state, we need a list of the languages spoken there.

The list is endless! Basically, we need these databases to help fill in forms for various things. And all these databases need to be internationalised too.

Another area where work is required is regarding local customs. For example, a widow in the West would wear black, whereas a widow in India would wear white. So the mourning colour here is white as opposed to black. An application that uses colour for this purpose would have *html* code like `<colour:"mourning colour"/>`, and the application would check the locale and choose the appropriate colour.

As for numbers, for us, 'a hundred thousand' is one lakh, a 100 lakhs is one crore, 100 crores is one arab and 100 arabs is one kharab—this should be transparently done in all programming languages.

As for calendars, we have a large number of them. The government follows the Gregorian calendar for all official purposes, but for many non-official purposes, the local calendars have to be followed. In fact, in Tamil Nadu, the revenue department records dates in both the Gregorian and the local calendar.

Regarding astronomy and astrology—there are many local versions of these and are needed in applications dealing with these subjects.

And maybe we even need a database on local uses of language or common expressions. In the West, to say a place is not far away, we say: 'a stone's throw away', whereas in Tamil, that would be '*koopidum dooram*', meaning close enough for your shout to be heard—we Indians do not throw stones to measure distance!

l10n is not rocket science. It is hard work, but anyone can do it if there is an interest. It is necessary work and needs to be done if the benefits of IT are to reach the general public. 

Kenneth Gonsalves works with NRC-FOSS at AU-KBC, MIT, Chennai. He can be reached at lawgon@au-kbc.org



How to Contribute to Open Source

There's a huge pool of open source software projects out there that you can contribute to. However, for those unfamiliar with the OSS development model, it's a bit daunting to even think where to get started. This article guides you on some of the steps in layman's terms, before explaining the kernel development cycle, briefly.

The open source community has a number of projects being developed around the world. Contributing to these open source software—applications and tools, including the kernel—is fun in a way, but there are still a number of tacit dos and don'ts for a newbie. In this article, we'll consider a few of them.

Some philosophy

The first and foremost part we need to consider here is the way in which the OSS projects operate. According to Dr Marietta Baba of Michigan State University, "Open Source development violates almost all known management theories."

Unlike the typical proprietary projects, the OSS projects do not start with the big 'design documents' but the development models being followed are incremental. The individual interests and company desires typically

take a back seat when it comes to project quality. The OSS model also teaches individuals to respect the interests of other people and exploit the synergies.

Some basic common steps

The following are some basic steps to consider when one decides to contribute to a FOSS project:

- *Find the correct home page/development wiki page:* Almost all the OSS projects have their own home pages with 'developer' corners or wikis. They typically also have sections on news, FAQs, documentation, downloads and the three most important links, namely, the mainline svn (or the relevant repository), the bugs (or 'known issues') section and the mailing list. This is all one needs to start with. The popular projects (like GCC) have IRC links and contributors could be reached there as well.
- *Join and work in the correct mailing list:* It is

actually a good idea to stay subscribed to these mailing lists before actually starting off with your own development. Don't be surprised if you start getting hundreds of mails a day, once you join one of these lists. As long as we behave ourselves on these mailing lists, people are pretty friendly and helpful.

- **The repositories and version control:** Typically, CVS, SVN, or GIT is selected as the version control system for most of the OSS projects. Take a while and familiarise yourself with the relevant one.
- **Follow the conventions:** All the development communities follow some conventions regarding:
 - Error handling
 - Project specific abstractions and encapsulations
 - Scalability
 - Portability and platform
 - Commenting/coding conventions
 The documentation regarding them is available in one form or the other with the project and has to be given due respect.
- **Bugzilla:** It is a great value addition—not directly in the development, but this is where users of your application file bugs. Bugzillas are maintained by almost all OSS projects and the project developers/maintainers respond to any bugs filed as long as the bugs are qualified with relevant data.
- **Documentation and FAQs:** Although this looks very trivial, it is an important part of any project—both for users and other developers.
- **Licensing policy:** Last but not the least, most OSS projects are governed by pretty strict licences. There are a number of open source licences (refer to www.opensource.org/licenses), out of which GPL is perhaps the most popular. Although the licences could contain a lot of legal jargon difficult for a developer to understand, you can always ask the project maintainers and mentors to explain to you the terms and conditions so that you can best adhere to the project licence as a contributor to a project.

Here comes the kernel

Like any other OSS software, the Linux kernel also follows an incremental development model. As Linus Torvalds, creator of Linux kernel, has said: Linux is an 'evolution' and not an intelligent design. The same backbone kernel pertains to multiple deployment scenarios ranging from cell phones with real-time requirements to trivial desktops/laptops and even to gigantic data centre servers.

The Linux kernel development process is very interesting and exciting to look at.

The kernel development takes place in 'development cycles'. Each 'development cycle' starts with a 'merge window'. During the merge window, code that is sufficiently stable and accepted by the developer community will be added into the mainline kernel.

The merge window lasts for two weeks, and by the end

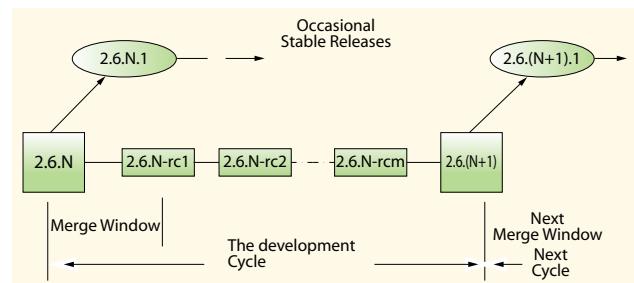


Figure 1: The development cycle of the Linux kernel

of two weeks, Linus Torvalds makes the first RC (release candidate) release, and this is followed by a number of RC releases for the next six to ten weeks. The main activity during this time frame is to 'stabilise' the code taken in during the 'merge' window.

After a sufficient number of 'release cycles', one more stable release is made and subsequently, the next merge window opens.

Parallel to this 'development cycle', there is a 'stable team' that looks after and maintains the 'stable release' given at the end of every development cycle. Figure 1 illustrates the complete process.

The repository used for the kernel development is Git, and the master mailing list for the kernel is at vger.kernel.org/vger-lists.html.

The code goes into the mainline kernel in the form of 'patches', and this 'patchwork' needs to be done in a timely manner—the respective developer has to follow a lot of conventions. The patches typically go first through one or more subsystem maintainer's tree before they become a part of the mainline kernel.

There are two more projects, namely "linux-next" and "mm", running in parallel, which support the main Linux kernel development.

For more detailed information on how to get started with kernel development, refer to ldn.linuxfoundation.org/book/how-participate-linux-community. 

REFERENCES:

- [How to contribute to the Linux kernel: ldn.linuxfoundation.org/book/how-participate-linux-community](http://ldn.linuxfoundation.org/book/how-participate-linux-community)
- [A recent interview with Linus Torvalds—Kernel contributions to Linux: news.zdnet.co.uk/software/0,100000121,39462454,00.htm?r=47](http://news.zdnet.co.uk/software/0,100000121,39462454,00.htm?r=47)
- [A slideshow of LCA 2008 talks: lwn.net/talks/lca2008/img0.html](http://lwn.net/talks/lca2008/img0.html)

By: Niles Govande. The author is a Linux enthusiast and could be contacted at nileshgovande@yahoo.com. His areas of interest include Linux system software, application development and virtualisation. He is currently working with the LSI Research & Development Centre, Pune.

Session Management Using PHP



Part 1: Cookie-based Sessions

Managing sessions protects Web pages from unauthorised access. It also provides website visitors the comfort of user-specific behaviour. This article explains session management strategies on a LAMP framework using PHP as well as the default session management capabilities that result from using cookies.

Most of us use sessions on a daily basis while browsing the Web. But have you ever wondered how sessions are implemented? The e-mail account we hold, the subscribed journal pages we read, the paid music channels we listen to—all these services use session management to identify their users. Session management provides two facilities: it protects the content from unauthorised access, and makes the same URL behave as per the requirements of the user.

Usually, most of us never care to think about what happens in the background when we enter our login credentials (user name and password). We probably consider this process quite trivial. In one of my recent projects to

build a login-based set-up, I needed to create sessions to provide measured rights to the users. The problem that I considered trivial, began to bog me down. I went through a few difficulties while creating reliable sessions, and in the process learnt some lessons. Let me share them with you in this article.

The project was on the back burner for a while for the want of a reliable session management strategy. Most times, PHP session management silently failed and I could not make head or tail out of the problem (later on, I found that disabling cookies in the browser made the PHP sessions fail). You can say I looked like Charlie Chaplin—doing all the serious work and coming out with laughable results. An implementation of sessions using a

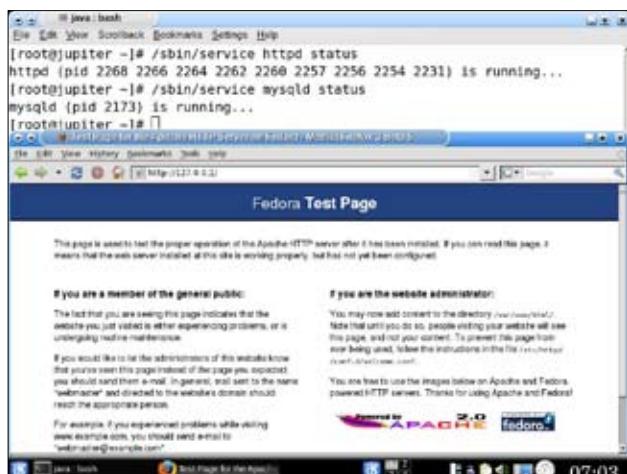


Figure 1: Testing Apache and MySQL services

server-side database worked correctly for me and I was able to successfully finish the project. The post-mortem of the 'failure of PHP sessions' showed that clients that block cookies cannot engage in a session.

What's in a session?

A session means the duration spent by a Web user from the time logged in to the time logged out—during this time the user can view protected content. Protected content means the information that is not open to everyone (like your e-mail inbox). The beauty of a session is that it keeps the login credentials of users until they log out, even if they move from one Web page to another, in the same Web service, of course.

From the point of view of a server-side programmer, the server verifies the user name and password obtained from the client and permits the client to create a session if the data is valid. On successful verification of login credentials, a session ID is created for the user, which should either be stored on the server or the client. Once the login information is stored, all subsequent pages identify the user and provide the requested information.

The two parts of this article (the second part will be published next month) explain two alternative strategies for creating sessions and explain the pros and cons of both. On the client side (about which we'll talk about in this issue), the session ID

can be stored in a small file called a cookie. The cookie stores a name, a value, the server from which it originated, the time of creation, expiry time, etc. This file is stored on the client machine with the permission of the browser. The browser settings affect the storage of cookies on the client machine.

Although the cookie-based strategy is simple, it comes with a few weak links. People might peep into the client machine, find the cookie and misuse the name-value pair to cheat the server, disguising themselves as authentic users. Hence, cookie-based sessions are not recommended for financial transactions, as they don't have the total assurance of privacy.

The second pitfall in cookie-based sessions is that a conservative user may block all incoming cookies. When the server sends a session cookie, it assumes that the client would store it. But, the client might reject the cookie for security reasons and thus hamper the formation of a session. This is what happened in my project. While some attempts to log in were successful, some were not. The reason being that blocking cookies was hampering sessions.

To make session management independent of cookies, a database was created on the server to store the session information. Each page of the site included a status-checking script, which queried the database, checked the validity of a session, and permitted further access only if the session was valid. This solution worked well, and I will present the procedure in the second part of this article.

Requirements for the project

I will use a LAMP stack to solve the session-management requirement. So, before proceeding into the

**ATTENTION
MCA, MSC (IT)
Final Year Students**

**GET Trained
Project Placed** 

- Projects available in LAMP, Java, J2EE, Perl, Python & Ruby On Rails
- Batch Discount available
- Accommodation will assist for out Station Students
- Register Now .. Limited Seats available
- IEEE Projects also available

Workshop on Weekends - Dec '08

Python, Django , Ajax, ROR
Jquery, Perl, Shell Scripting

For more details
www.lynusacademy.com/workshops

**Special Offer for
Job hunters &
Freshers Zend
PHP Training
Rs. 23,200/- Only**



Other courses

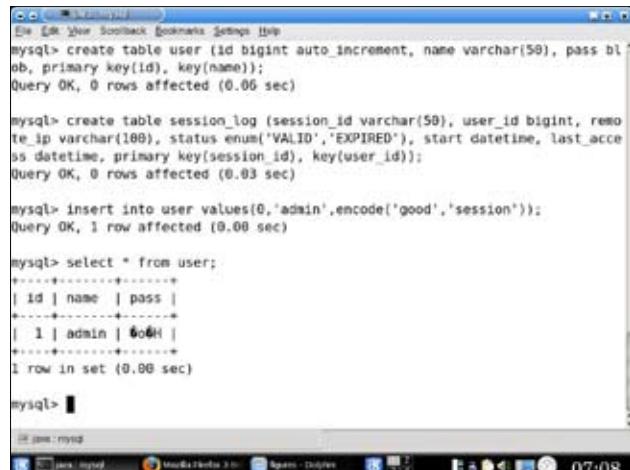
- RHCT & RHCE
- Shell Scripting
- Perl Scripting (for Sys. Admin)
- G-FOSS - Web Development (3 Months & 6 Months)
- G-FOSS - Engineer (3 Months & 6 Months)
- Perl + CGI Programming
- Python Programming
- Ruby on Rails
- AJAX & JQuery
- MySQL - for Developer
- MySQL - For Administrator
- PostgreSQL - for Developer
- PostgreSQL - for Administrator



Future is open

Lynus Academy Pvt Ltd

Chennai - 044-25021630,
42171278, 9952966527
Pondicherry - 0413-2276233,
4202121, 9952966528
contactus@lynusacademy.com
www.lynusacademy.com



```

mysql> create table user (id bigint auto_increment, name varchar(50), pass blob, primary key(id), key(name));
Query OK, 0 rows affected (0.06 sec)

mysql> create table session_log (session_id varchar(50), user_id bigint, remote_ip varchar(100), status enum('VALID','EXPIRED'), start datetime, last_accessed datetime, primary key(session_id), key(user_id));
Query OK, 0 rows affected (0.03 sec)

mysql> insert into user values(0,'admin',encode('good','session'));
Query OK, 1 row affected (0.08 sec)

mysql> select * from user;
+----+-----+-----+
| id | name | pass |
+----+-----+-----+
| 1 | admin | 6064 |
+----+-----+
1 row in set (0.00 sec)

mysql>

```

Figure 2: Creating the database tables for session management

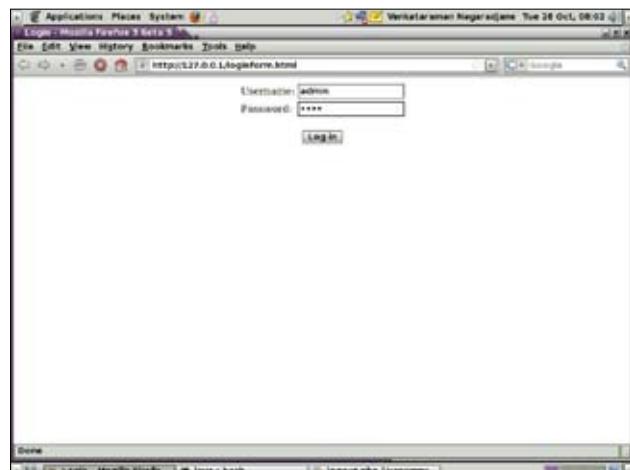


Figure 3: The login page

project, you must verify the availability of the required services on the machine. It is assumed that a working Linux installation is available. Few words on Apache and MySQL servers might make things easy. Checking Apache and MySQL servers requires root user privileges. Either use the `su` - or `sudo` command to obtain root user privileges before running the `/sbin/service` tool.

Both Apache and MySQL run in the background as daemons, and their status can be checked using the `/sbin/service` command. The server can be in any one of the three states on the machine: a) the service is available and running; b) the service is available on the system, but not running—starting the service is easy; c) the service is not available on the machine—download and install the required service.

The availability of the Apache server on a machine may be tested by issuing the command `/sbin/service httpd status` on the command window. The response for this command might say 'httpd is running', wherein we can proceed to the next stage. If the service is available on the machine, but not currently running, the response would say 'httpd is stopped'. No problem! Issue the command `/sbin/service httpd start`. The third possible response to `/sbin/service httpd status` command is

'httpd: unrecognized service', which means the Apache server needs to be installed on the machine.

Follow the same procedure to check the availability of the MySQL database server on the machine, after substituting `mysqld` for `httpd` in the previous command sequences. Figure 1 shows the command window and the browser window when the services are rightly installed on the system. Note that I've used a Fedora 9 installation for this article. If the `service` command is not the default command to start and stop services (daemons) on your system, please consult the documentation to find out the substituting command.

After ensuring that the services are available, check for the availability of the PHP scripting language by issuing the `php -version` command and look at the response. If the version number, build date, etc, are displayed, it means PHP is available. If a message stating "php: command not found" is displayed instead, then you need to install PHP, of course.

Once the Apache, MySQL and PHP servers are ready, you can readily run and test the scripts provided in this article. For those who use very old versions of the Apache server, PHP might require a few configurations; in recent versions, PHP does not require any further configurations.

Knowing how to work

For accessing content through a Web server, the data should be placed at the root of the Web server. In case of the Apache server, the documents for the server should be placed in the directory `/var/www/html/` (or any other directory specified as *DocumentRoot* in your `/etc/httpd/conf/httpd.conf` file). Remember that all the HTML and PHP files mentioned in this article reside at the *DocumentRoot*, which is the `/var/www/html/` directory in case of my Fedora 9 system. Placing any file in Apache's *DocumentRoot* permits everyone to access the page, by accessing the server's URL.

As for the database, at least one table is required for session management to store the user names and passwords to authorise login requests. Connect to your MySQL server by using the following command:

```
mysql -u <username> -p
```

This will prompt for the MySQL password. After providing the password, once you get the mysql prompt, you can create a database called 'session' by issuing the following command:

```
create table session; use session;
```

Following this, you can create a table to store the user name and password by issuing the following command:

```
create table user (id bigint auto_increment, name varchar(50), pass blob, primary key(id), key(name));
```

This table is enough for the first part of the example,

which handles sessions using a cookie.

For handling sessions using the database table on the server side (which we will deal with in Part 2 of this article), issue the following command at the mysql prompt:

```
create table session_log (session_id varchar(50), user_id bigint, remote_ip
varchar(100), status enum('VALID','EXPIRED'), start datetime, last_access
datetime, primary key(session_id).key(user_id));
```

This will create the table required to store the session information. Figure 2 shows the commands as issued through the MySQL command line.

After creating the tables, insert at least one user into the user table. The typical command I used for the test case is as follows:

```
insert into user values(0,'admin',encode('good','session'));
```

This command inserts a user named 'admin' with the password 'good', encoded using the key *session*. Calling *decode* for the password requires the same key for a correct retrieval of the password.

Cookie-based sessions

PHP provides a cookie-based implementation for session management. The *\$_SESSION* array is used for storing session data. PHP automatically generates a session ID and sends a session cookie containing this session ID to the client machine. The PHP functions for session management are listed in Table 1.

The basic login process begins with the display of two fields for the user name and password. The following code shows the HTML file used for the display of the login prompt (Figure 3 shows the login page):

```
<html><head><title>Login</title></head>
<body>
<form method="post" action="login.php">
<center>
<table border=0>
<tr>
<td><label for="username">Username:</label></td>
<td><input type="text" id="username" name="username" maxlength="50"
/></td>
</tr> <tr>
<td><label for="passwd">Password:</label></td>
<td><input type="password" id="passwd" name="passwd" /></td>
</tr>
</table><br />
<input type="submit" value="Log in" />
</form>
</body></html>
```

Logging in

The user name and password are passed to the PHP script called *login.php*. The script uses the *\$_POST* global

TABLE 1: SESSION MANAGEMENT FUNCTIONS

Function	Purpose
<i>session_start()</i>	Initialises a session. If the session was not already started, it sends the session cookie to the client machine. If the session was already started, it loads the <i>\$_SESSION</i> global variable with whatever values it was previously initialised with.
<i>session_destroy()</i>	Destroys the session. The variable <i>\$_SESSION</i> is cleared and the session cookie on the client is killed.



Figure 4: The session cookie (PHPSESSID) is displayed in Firefox's cookie list

variable to get the values filled against the user name and password fields. A connection is then established with the database 'session' and the corresponding user's ID and password are retrieved. If the user name is found, the password stored in the database is matched against the password supplied by the user. If they do not match, the login attempt is rejected. Otherwise, the login succeeds. The following is what the *login.php* file looks like:

```
<?php
/*login.php*/
function check_login($username, $password)
{
if (!($username && $password))
    return false;
$conn = mysql_connect('127.0.0.1', 'root', 'sivasiva') or die(mysql_error());
mysql_select_db('session', $conn) or die(mysql_error());

$result = mysql_query("SELECT id, decode(pass,'session') as pass FROM
user WHERE name=".addslashes($username)."'", $conn) or die(mysql_error());

if(mysql_num_rows($result) != 1) {
    printf("<center><h1>User %s not found</h1><br /><a
href=\"login.html\">Go to login page</a></center>", $username);
    return false;
}
```

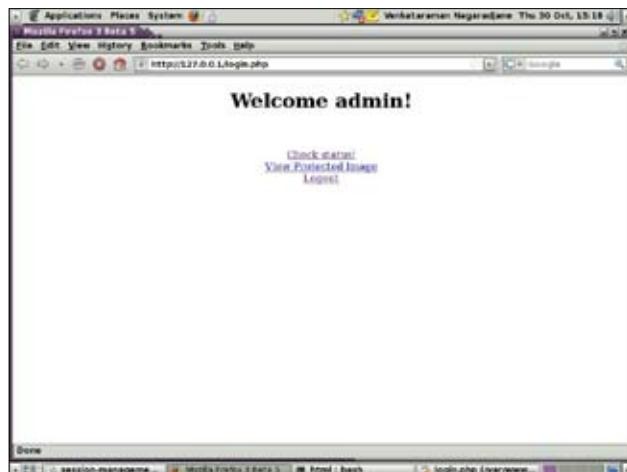


Figure 5: The welcome screen upon successful login



Figure 6: This image is protected content



Figure 7: Access to protected content denied when session is not available

```

        }

if(mysql_result($result, 0, 'pass') != $password) {
    printf("<center><h1>Login attempt rejected for %s!</h1><br /><a href=\"login.html\">Go to login page</a></center>", $username);
    return false;
}

session_start();

```

```

$_SESSION['username'] = $username;
$_SESSION['id'] = mysql_result($result, 0, 'id');
mysql_close($conn);
return true;
}

if(check_login($_POST['username'], $_POST['passwd'])) {
    printf("<center><h1>Welcome %s!</h1><br /><br />\n".$_SESSION['username']);
    print("<a href=\"status.php\">Check status!</a><br /><a href=\"protectedimage.php\">View Protected Image</a><br /><a href=\"logout.php\">Logout</a></center>\n");
}

```

In case the user name and password are correct, the `session_start()` function is called, which, in turn, sends a session cookie containing the session ID of the user to the client machine. The cookie is shown in Figure 4. After this, calling `$_SESSION['username']` or `$_SESSION['id']` is permitted to store and retrieve session data. In the present case, the user name and user ID are stored in the `$_SESSION` array.

The session ID created by the `session_start` function is stored in a cookie on the client machine. You can inspect the cookie by accessing `Edit-->Preferences` from the Firefox menu, selecting the 'Privacy' tab, followed by clicking the 'Show Cookies' button. This displays the cookies sorted by the name of the server. In the present case, the server resides at `127.0.0.1` and the cookie is called 'PHPSESSID'--you can notice this value displayed against 'Content' field on the information area. The welcome screen displayed on login is shown in Figure 5.

Session status

Since the session has been established, you can test the availability of persistence for the user name and user ID. For this, let's create a small script called `status.php`. This script calls the `session_start()` function. Since the session cookie is already available in the client machine, calling the `session_start()` function looks at the session ID and loads the appropriate session variables with previous values on the server machine. Hence, calling `$_SESSION['username']` or `$_SESSION['id']` will retrieve the data stored through the `login.php` script file. The following is what the `status.php` script looks like:

```

<?php
/*status.php*/
session_start();
//Check for valid session. Exit from page if not valid.
if(!isset($_SESSION['username']) && !isset($_SESSION['id'])) {
    print("<center><h1>invalid session!</h1><br />\n<a href=\"loginform.html\">Login</a>");
    exit();
}
printf("<center><b>Welcome %s! Your id is: %d</b>%s", $_SESSION['username'], $_SESSION['id']);

```

```

SESSION['username'].$_SESSION['id']);
printf("<br /><a href='logout.php'>Logout %s</a></center>",$_
SESSION['username']);
?>
  
```

The status script can be accessed by clicking on the 'Check Status' link on the login page. It displays the user name and user ID obtained from the session data. This fulfills the basic requirement of a session, since it permits persistence of data across different pages after login.

Creating protected pages

The very aim of a session is to create protected pages. A simple PHP script file is listed below that will protect an image from public access. The PHP script file *protectedimage.php* calls *require_once('status.php')* at the very beginning. This executes the status script once. The status script finds out the validity of the session, permits further movement if the session is valid and calls *exit* otherwise. The code for the protected image is shown below:

```

<?php
/*protectedimage.php*/
require_once('status.php');
print("<div align='center'><img src='summer.jpg' width=75%></
div>");
```

The protected image that is displayed after proper login is shown in Figure 6, while the same page (*http://127.0.0.1/protectedimage.php*) loaded without a valid session is shown in Figure 7. Look at the URL on the address bar of the browser in both the figures—the same URL displays the image when the session is available and denies access when the session is not available.

Log out

Now that we have checked how protected content works, it's time to script the logout operation. The logout operation is contained in the script called *logout.php*. The script calls the *session_destroy()* function, which kills the session cookie and clears the session variables. The logout screen is shown in Figure 8. The following is what the logout script looks like:

```

<?php
session_start();
printf("<center><h1>Good Bye %s!</h1><br /><a href='loginform.
html'>Go to login page!</a><br /><a href='status.php'>Get Status</
a></center><n>$_SESSION['username']);
session_destroy();
?>
  
```

We might test whether the session was really terminated by calling the *status.php* file to check whether the name or



Figure 8: Logout screen



Figure 9: Status message after destroying the session

the ID are still available. Figure 9 shows the message that the session is invalid. The name and ID are not available after destroying the session. Hence, including *status.php* at the beginning of each protected page ensures access is only possible after proper login, otherwise all other requests to the URL get terminated at the beginning itself.

Pros and cons of cookie-based sessions

Cookie-based session management provides the easiest way to manage sessions, especially since PHP provides built-in capabilities for this. However, there is also a strong reason why it should be avoided for professional websites because if the browser is set to block cookies, cookie-based session management fails. Another pitfall is that the cookie might fall into mischievous hands and result in loss of information. Hence, a cookie-based session is useful only for non-monetary and non-confidential websites.

The second part of this article (to be published next month) will explain server-side sessions using database tables.



By: V. Nagaradjane. The author is a freelance programmer and can be contacted at nagaradjanev@rediffmail.com



For Aspiring Game Designers

The Allegro library makes it all very simple.

Designing and programming games is anything but easy. One needs to understand the game logic (algorithm) and the graphics manipulation techniques—together, they form the 'core' of a game—to design a game. And everyone agrees that the graphics alone make a huge impact on games.

To begin programming the games and their graphics we need to have the basic concepts of game programming clear, as well as a good graphics and I/O tool that does not drag a beginner into the complex world of syntaxes, data structures, procedures, complex internals, etc. Thus, I'm sure everyone agrees that starting with OpenGL or DirectX programming becomes quite a job for beginners.

In this article we will introduce you to a 2D and 3D game and a graphics library called Allegro, primarily to be used with the C programming language, which brings you a great platform to start game programming.

Although you will still need to know the basic techniques and algorithms to design the core, Allegro takes good care of the graphics, sound, I/O and all the other components. So you can put most of your efforts to design a great core, and then create the multimedia components, and the I/O using Allegro with great simplicity. Allegro does this by hiding the complex internals with its simple abstract data structures and similarly simple routines. And all this is not just for beginners -- this library also has the power for advanced and professional-level programming.

From the history books

Allegro used to stand for Atari Low LEvel Game Routines when it was written from scrap in C, by Shawn Hergreaves for the 1985's PC platform Atari ST. However, over time the Atari ST platform was discontinued and thus the development for Allegro also stopped, only to restart again in 1995 with Borland C++ and DJGPP compilers. When the

development again stalled with Hergreaves getting more involved in other important work of his own, interested people came forward and kept the project going.

Allegro now stands for Allegro Low LEvel Game Routines, a recursive acronym, and the graphics and game library is primarily to be used with C and C++ languages. It indeed is a free software; however, the project, headquartered at alleg.sourceforge.net, likes to call it 'giftware' that gives you the right to "...use, modify, redistribute, and generally hack about in any way you like, and you do not have to give us [the developer community] anything in return." The latest 'work-in-progress' (unstable) release of Allegro is version 4.9.6. Released on November 2, 2008, this will go on to become version 5 through updates. In this article we will be discussing Allegro 4.2.2.

What's in it?

To design a game, different sets of routines are needed that cover different fields of the design—graphics being one of the most important fields. 2D graphics is covered very well in Allegro with basic graphics routines like pixel, line, circle, arc, rectangle, triangle, z-buffered polygon, bezier spline, flood fill, etc, along with their different variations that bring in flexibility and ease of use. A distinctive feature is that Allegro has callback functions for each basic shape, which let you draw that shape abstractly.

Loading image formats like BMP, TGA, PCX and LBM are natively supported. Other formats can be loaded after suitable plug-ins are installed. PNG and JPEG support is added in v4.3.10. The loaded images can be resized, and effects like lighting and blending can be added too. Also supported are features like blitting, direct draw to screen or bitmap, and clipping.

A separate set of sprite handling routines allows you to include bitmap sprites and manipulate them (pivot, flip, rotate, scale, stretch, etc). Support for transparent and animated sprites packs more power in sprite programming. Support for Run-length-encoded (RLE) sprite and compiled sprite lets you balance the performance and size of the game. Reduction, alpha blending, gouraud shading of sprites and bitmaps allow you to add more effects. Allegro also lets you directly access the video memory.

3D graphics routines of Allegro enable you to draw basic 3D polygons and render them with MMX/SSE/ 3DNow! extensions. Although Allegro doesn't offer much variety and gives no hardware acceleration with its 3D graphics routines, all these can be implemented along with proper mathematical manipulations to make some 3D graphics. This drawback is countered by the AllegroGL library that enables you to make OpenGL-rendered graphics with a separate set of AllegroGL's 3D routines to use with OpenGL coordinates. Double/triple buffering, scene rendering, hardware scrolling, and mode-X split screen are also supported natively.

Realistic games need a physics engine that drives the different game objects based on physical laws. Sounds cool,

but they are actually mathematical manipulations done as per different physical laws to create the proper graphic elements, which is why there's the need for graphic-oriented math routines. The real headache is to write these routines efficiently. Allegro comes with the medicine for this headache with its own math routines containing fixed-point trigonometric and algebraic functions, 3D math routines that have almost all required matrix and vector operations, and manipulation routines and quaternion math routines, for easier rotations and accurate interpolations.

Allegro supports 32bpp (over 16 million) colours, as well as lower depths (8, 15, 16, 24bpp) with an amazing 1600 x 1200 resolution. Conversion between RGB and HSV colour formats is also supported. Transparency effects, patterns drawing and alpha, colour, burn and many different blend effects and palettes can also be used.

Obviously, Allegro can benefit from the graphics driver that's installed. There are various graphics drivers that are supported by Allegro:

- X Window, DGA, fbcon, SVGAlib, VBE/AF, mode-X, VGA drivers under UNIX
- GDI, DirectX with full screen, under Windows
- Quartz under MaxOS X
- BwindowScreen and BDIRECTWindow under BeOS
- VGA 13h mode, mode-X, and SVGA up to 32bpp under DOS

In the current unstable releases, that will eventually become Allegro 5.0, AllegroGL has been included in addition to an OpenGL and Direct3D driver.

The FreeBE/AF project at www.talula.demon.co.uk/freebe is a free implementation of accelerated VBE/AF drivers, which adds portable 2D hardware acceleration under a number of cards.

But a game is not only about the graphics; the sound needs to be equally good. Similar to graphics, the sound detection in Allegro is automatic, and it can work with your currently installed sound driver. Digital sample routines natively support VOC and WAV file formats to be loaded and played directly (OGG support has been added in v4.3.10), and add basic effects like echo, frequency up/down, volume up/down, vibrato, pan, sweep, etc. Additional sound libraries let you load other file formats. Allegro can play MIDI files and lets you control the notes, panning, pitch, bend, loop, drum mappings, etc. You can apply up to 64 effects simultaneously. Audio streaming routines help you play huge audio file sizes, as well as streaming audio.

What about an online multi-player game where players communicate through voice commands? No issue really! Allegro has sound recording routines that lets you make voice commands on a multi-player game, or implement voice recognition in your program.

Finally, the sound drivers supported by Allegro are:

- OSS, ALSA, ESD, aRTs, JACK and SGI AL under UNIX
- Direct Sound WaveOut and MIDI drivers under Windows.
- CoreAudio, Carbon Sound Manager and QuickTime Note

Allocator under MacOS X

- Adlib, Sound Blaster and its variants, AudioDrive, Ensoniq, etc, under DOS
- BSoundPlayer and BmidiSynth under BeOS, and more

The most essential thing in game play is inputs from the player's end. Programming inputs through keyboard and mouse with Allegro is quite easy using the in-built global variables that get you all the parameters and values you need. The keyboard and mouse are auto detected and configured with only a function call. Even joysticks can be programmed with the same ease after proper calibration. Allegro comes with advanced high-resolution timers and interrupts, which let you finely control game play with efficient programming. It also has multi-threading features.

A game is never complete without a good interactive GUI, which lets the users select options from interactive menus. To serve this need, Allegro also has an object-oriented dialog manager, with text boxes, push buttons, radio buttons, check boxes, menus, file browsers and all the essential objects that use the same keyboard and mouse inputs.

How about a game story video, or a cut scene in your game? Allegro has a very simple solution to this matter as well, with its range of FLIC routines that play Autodesk's FLI and FLC animations with only a line of code.

Allegro gives you many options for printing texts and fonts. Although the default text format is UTF-8, it also supports other encoding formats and conversion between them. The different text functions and their variations let you place text in any position of the screen without much calculation. You can use GRX or BIOS (.fnt) fonts, as well as fonts from bitmap images and other sources. Installing an additional library enables Allegro to support True Type Fonts (ttf). TTF support was added in the work-in-progress release v4.9.4.

Allegro's configuration routines help saving and loading different configurations, including hardware settings.

Now, analyse this: a game needs sprites, level maps, files, animations and other important game data. Distributing a game executable with separate bitmaps, sounds and movies all unencrypted and uncompressed would be difficult to manage and update, as the size and number of the files increases, and would also be insecure. A third-party compression/decompression/encryption library for the data seems a possible solution. But this comes with the hassle of finding and installing a proper one, in addition to learning it. All this trouble vanishes when you are introduced to Allegro's Grabber utility, and datafile feature. They let you pack your sprites, sound, animations, level maps, and all other types of files into one single compressed datafile, which can optionally also be encrypted. Allegro datafiles use the LZSS compression algorithm. These datafiles bring in great flexibility in creating games. For example, separate datafiles containing different levels of data are easy to distribute and to develop without altering other levels, in addition to saving bandwidth and making

less development mess. To manage these datafiles from inside the program, Allegro has file and compression routines.

Writing programs with Allegro becomes easier because of the variety and simplicity of inbuilt predefined types, structures, and global variables. For example, the 'mouse_x' and 'mouse_y' global variables give you the mouse positions on screen at any time. The most fascinating feature of Allegro is that it provides callback functions to almost every basic function. This allows great control and flexibility and lets you program an abstract variant of a certain function. Like with the callback variant of the 'line', with the 'do_line' function you can draw an abstract linear formation with the pixels arranged as you like.

Perhaps the best feature of Allegro is that it is cross-platform, which lets your program run in almost all the popular OSs and hardware platforms, and your code to be compiled in a very wide range of compilers. So you don't have to worry about modifying your code to support different environments. A source code once written could be compiled under any OS, with any compiler (with Allegro installed) without a single modification in the program, avoiding the huge amount of conditional compilation pre-processor directives and related complications.

It supports all major OSs like DOS, UNIX, Linux, FreeBSD, Darwin, Irix, Solaris, Windows, BeOS, QNX, MacOS X, etc. As for compilers, it supports DJGPP, GCC, Microsoft Visual C++, Borland C++ /C++ Builder, Dev C++, MinGW32/Cygwin, Digital Mars Compilers, etc.

This library is basically a C/ C++ add-on library. For those who are not into C/C++, there's no reason to be disappointed—other language binding editions like Python, Perl and Pascal are also available. Check alleg.sourceforge.net/bindings.html for more information on how to use Allegro with other languages.

Phew! That's a long list of features offered by Allegro, isn't it? Guess you'd now like to know how to get started.

Getting your system ready

Visit alleg.sourceforge.net/wip.html and download the latest stable release—*allegro-4.2.2.targz*. Scroll down the page to get non-Linux downloads. We will all install the AllegroGL library, so that OpenGL support is available. For that, visit allegrogl.sourceforge.net/wiki/Download and click the download link to download the latest version—*allegogl-0.4.3.tar.bz2*.

Assuming you are installing the library from source in Linux, with GCC, extract the contents of *allegro-4.2.2.targz* in a directory and *cd* into the extracted directory as follows:

```
$ tar xvfz allegro-4.2.2.tar.gz
$ cd allegro-4.2.2
```

Now, you need to convert the files to the UNIX format and set the proper makefile:

```
$ chmod +x fix.sh ### in case it is not executable
$ ./fix.sh unix
```

Now configure and make the library—the passed parameter lets you create both dynamic and static executables:

```
$ ./configure --enable-static=yes
$ make
```

Become the root user and install the library into your system:

```
$ su -
# make install
```

Next, open the */etc/ld.so.conf* file and append the line */usr/local/lib* in it. Run */sbin/ldconfig* to load the update.

To make offline documentations, execute the following:

```
# make install-man
# make install-info
```

To generate a 475-page handy documentation file in the allegro-4.2.2/doc/ directory execute:

```
# make docs-pdf      ### for a PDF
# make docs-dvi      ### for a DVI file
# make docs-ps      ### for postscript file
```

Now *cd* into the set-up directory and run the “setup” executable to configure Allegro for your system:

```
$ cd setup
$ ./setup
```

This step is not very important, as Allegro autodetects hardware configuration.

Now, to install the AllegroGL library, extract the contents of the package (*allegl-0.4.3.tar.bz2*), and *cd* into extracted directory:

```
$ bzip2 -d allegl-0.4.3.tar.bz2
$ tar -xvf allegl-0.4.3.tar
$ cd allegl
```

Set the makefile for UNIX, configure and then make:

```
$ chmod +x fix.sh
$ ./fix.sh unix
$ ./configure
$ make
```

Become the superuser, and install the library:

```
# make install
```

And that's the end of Allegro and AllegroGL installation.

Oh, and in case you do not like installing things from source, have a look at your distribution's software repository; chances are that it's also available there. And just in case you want to test the unstable release, you can install it in a similar way by using *cmake* instead to ‘build’—of course, after downloading the appropriate package. Note that you don't need to install AllegroGL manually in this case, as it is inbuilt in v4.3.10 or later releases.

Getting started

To get you started and run a program right away, we will present some samples. Let's begin with the following:

```
#include<allegro.h>

#define RED makecol(255,0,0)
#define GREEN makecol(0,255,0)
#define BLUE makecol(0,0,255)
#define BLACK makecol(0,0,0)
#define WHITE makecol(255,255,255)

int main(void)
{
    allegro_init();
    install_keyboard();
    install_mouse();
    install_timer();

    allegro_message("This is the first allegro program (Press OK)");
    set_color_depth(32);
    if(set_gfx_mode(GFX_AUTODETECT_WINDOWED,800,600,0,0))
    {
        allegro_message(allegro_error);
        exit(0);
    }

    textprintf(screen,font,10,10,WHITE,"Screen Resolution %dx%d",SCREEN_W,SCREEN_H);
    rect(screen,20,20,300,130,GREEN);
    circle(screen,SCREEN_W/2,SCREEN_H/2,50,WHITE);
    putpixel(screen,SCREEN_W/2,SCREEN_H/2,RED);
    line(screen,300,300,200,200,GREEN);
    triangle(screen,500,500,500,550,300,550,BLUE);

    show_mouse(screen);

    while(!key[KEY_ESC])
    {
        textprintf(screen,font,30,30,RED,"Mouse Pos x:y=%3d:%3d",mouse_x, mouse_y);
        textprintf(screen,font,30,50,BLUE,"Mouse Scroll Pos: %3d",mouse_z);
        textprintf(screen,font,30,70,GREEN,"Mouse Left Button Pressed:%3s",((mouse_b&1)? "Yes": "No"));
        textprintf(screen,font,30,90,GREEN,"Mouse Right Button Pressed:%3s",((mouse_b&2)? "Yes": "No"));
        textprintf(screen,font,30,110,GREEN,"Mouse Middle Button Pressed:%3s",((mouse_b&4)? "Yes": "No"));
    }
}
```

```
b&4)?Yes":No"));
rest(5);
}

allegro_exit();
return 0;
}
```

Although the above program is more or less self-descriptive, I'll still try to explain it in places. To begin with, before doing anything you need to include the *allegro.h* header file. The *makecol* function converts the values passed for colours as parameters to pixel format as per the current video mode requirements.

Before we call any Allegro function, we need to initialise it with the *allegro_init()* function. The keyboard and mouse are initialised in line 9 and 10. In line 11, we initialise the timer. The *allegro_message()* function prints any formatted text that has been passed in a dialogue box. Before we initialise the graphics driver we also define the colour depth we will use. We set it 8 here in line 13.

The *set_gfx_mode* function is used to initialise the graphics with the first parameter describing the graphics driver (here it is auto detected with the *GFX_AUTODETECT_WINDOWED* macro) and set the screen resolution (here it is 800x600 pixels). This function returns zero on success.

The next set of statements is self-descriptive, drawing the basic shapes with Cartesian coordinate values and colour values. Note that each such draw function has a 'screen' at its beginning. It's a global bitmap pointer of the current screen. The first parameters denote the destination bitmap where we are going to draw. Here we draw on screen, so the bitmap pointer is 'screen'. The function in line 25 shows the mouse pointer on screen. The global 'key' array contains a series of flags indicating the states of each key. We can check the key status of the alphanumeric keys using 'KEY_A'.. 'KEY_Z' 'KEY_0'.. 'KEY_9' as the index in the array *key* (check the manual for all keys).

The *while* loop executes while the *Esc* key is not pressed. In the loop the three *textprintf*s functions print the screen resolution, mouse cursor position, the mouse button and scroll status. They are updated in each iteration. The coordinates of the mouse cursor are accessed through the values of global variables: *mouse_x* and *mouse_y*. The scroll wheel position is retrieved with the *mouse_z* variable. The variable *mouse_b* represents the mouse clicks depending on the value it contains. Left, right and middle click is represented with values 1, 2, and 4, respectively. The *allegro_exit()* function closes the allegro system.

We now make a program to load a *.tga*, *.pcx*, or *.bmp* image:

```
#include<allegro.h>

int main(int argc, char *argv[])

```

```
{
BITMAP *image;

if(argc==1)
{
printf("\nUsage: program_name /path/to/tga/pic\n");
exit(0);
}

allegro_init();
install_keyboard();
set_color_depth(32);
if(set_gfx_mode(GFX_AUTODETECT_WINDOWED,800,600,0,0))
{
allegro_message(allegro_error);
exit(0);
}

image=load_bitmap(argv[1],NULL);
if(!image)
{
allegro_message("Error Loading%s",argv[1]);
return 1;
}

blit(image,screen,0,0,0,SCREEN_W,SCREEN_H);
//stretch_blit(image,screen,0,0,image->w,image->h,0,0,SCREEN_W,SCREEN_H);
while(!key(KEY_ESC));
destroy_bitmap(image);
allegro_exit();
return 0;
}
```

Note that we have defined 32-bit colour to show the picture properly. The last *while* loop waits for an Esc to be pressed, so we can see the picture. This program first tries to load the image passed through the command line as a BITMAP variable *image*. It also checks whether it was able to load it successfully. The *blit* function copies the rectangular area starting at screen position 0,0 with the width and height of the window (which we get from *SCREEN_W* and *SCREEN_H*) from the bitmap 'image' to bitmap 'screen'. The commented statement with the *stretch_blit* function is an advanced version of *blit*, which supports resizing the source bitmap.

At last we present a code with which you can drive a shape with the arrow keys around the screen:

```
#include<allegro.h>

#define MAX 50
#define MIN -50

int main(void)
{
int x,y,dx=0,dy=0;
```

```

allegro_init();
install_keyboard();
install_timer();

set_color_depth(8);

if(set_gfx_mode(GFX_AUTODETECT_WINDOWED1280,1024,0,0))
{
    allegro_message(allegro_error);
    return 1;
}

x=30,y=SCREEN_H-30;

while(!key[KEY_ESC])
{
    if(key[KEY_UP])          dy--;
    if(key[KEY_DOWN])        dy++;
    if(key[KEY_LEFT])        dx--;
    if(key[KEY_RIGHT])       dx++;

    if(dx>MAX)  dx=MAX;
    if(dy>MAX)  dy=MAX;
    if(dx<MIN)  dx=MIN;
    if(dy<MIN)  dy=MIN;

    x+=dx/15;
    y+=dy/15;

    if(x>SCREEN_W-30)
        (x=SCREEN_W-30;dx=0);
    if(y>SCREEN_H-30)
        (y=SCREEN_H-30;dy=0);
    if(x<30)
        (x=30;dx=0);
    if(y<30)
        (y=30;dy=0);

    textprintf(screen,font,15,15,15,"x=[%3d] y=[%3d] dx=[%3d] dy=[%3d]",x,y,dx,dy);
    circlefill(screen,x,y,10,15);
    vsync();
    circlefill(screen,x,y,10,0);
}

allegro_exit();
return 0;
}

```

This code is left for the readers to figure out.

Check out the *examples* directory inside your extracted Allegro tar ball for some excellent examples, and refer the allegro manual to consult each and every function.

Let's compile

To compile a program with Allegro routines we need to link it to proper libraries. Allegro supplies a program called *allegro-config* that's created and installed while installing

the library, which does all these mundane tasks on your behalf. To compile a program, you need to execute the following:

```
$ gcc source_code.c `allegro-config --libs`
```

...or:

```
$ gcc source_code.c `allegro-config --shared`
```

...or:

```
$ gcc source_code.c `allegro-config --static`
```

Please note that those are not single quotes in the above commands, but back ticks (the key below the ESC button). Any of these basic commands creates an *a.out* executable file. The difference is: while running the first and second command makes a shared executable file, the third one makes a static executable file.

You can, of course, include other GCC options as per your need. You can execute *allegro-config* without any options in the terminal to look at more linking options. And to know what libraries are linked, execute *allegro-config --libs*.

To compile an AllegroGL program (with Allegro 4.2.2) you need to additionally link *-lagl -lGL* and *-lGLU*, along with the previously mentioned compilation command. So, it becomes:

```
$ gcc source_code.c `allegro-config --libs` -lagl -lGL -lGLU
```

Check wiki.allegro.cc/Category:IDE_configuration to configure Allegro for your IDE.

Different platform-specific details could be found at the 'Platform specifics' section of the manual from www.allegro.cc/manual and docs/build in Allegro's extracted directory.

Now, what are you waiting for? Go start coding a top-scrolling space shooter!

Going further

Apart from Allegro's inbuilt tools (the grabber utility, colour mapping, text conversion, etc), there are other tools, utilities, and add-on libraries available that add more power to it.

To get more image and sound formats, and many more libraries and tools, check www.allegro.cc/resource/Libraries and www.allegro.cc/resource/Tools.

Utilities like Allegro Sprite Editor, and Allegro Font Editor help you to make your fonts and sprites, and they can be downloaded from www.allegro.cc/depot/utility/listing.

Information on Mappy, a map editor that has also been used to make commercial games, is available at [www.tilemap.co.uk](http://tilemap.co.uk).

What about help and tips?

Allegro has excellent documentation. Apart from the documentation we built and installed while installing the library, additional documentation is available for download at alleg.sourceforge.net/api.html. Almost all official documentation is included in the downloaded packages, so I would ask you to check those first under the *docs/*, *examples/*, *tools/* and the sub-directories inside the extracted Allegro directory.

Download the Allegro vivace tutorial from www.glost.eclipse.co.uk/gfoot/vivace. This is a great tutorial which teaches different game programming techniques in a very organised manner with lots of examples.

The official FAQ is at alleg.sourceforge.net/faq.html (or *docs/html/faq.html* in the extracted directory). As for tutorials, check alleg.sourceforge.net/docs.html and www.allegro.cc/resource/HelpDocuments.

Allegro.cc is the major unofficial Allegro website, which supplies you with compilers, add-on libraries, tools, Allegro games and more. The Allegro.cc forum [www.allegro.cc/forums] is a small, yet an active forum where you can venture in times of doubt.

As for AllegroGL, it also comes with well organised offline documentation. Check *docs/html/index.html* inside the extracted AllegroGL directory. For more information and details, go through the *readme.txt*, *howto.txt*, and *quickstart.txt* files inside the AllegroGL extracted directory. To check out online tutorials, point your browser to allegrogl.sourceforge.net/wiki/Resources.

Additionally, you can check the Allegro wiki at wiki.allegro.cc for more articles, and visit nehe.gamedev.net and www.swiftless.com for OpenGL programming tutorials.

And if you are still hungry to read more you might as well read the feature at www.gamasutra.com/features/19991026/allegro_01.htm.

This documentation is more than enough if one is already introduced to basic game design techniques and has some experience with C programming. For a newcomer to game design, you need to understand the game design logic and the algorithms, for which I would refer a book that teaches game design with Allegro and AllegroGL library, called *Game Programming All In One* (Latest 3rd Edition) by Jonathan S. Harbour. This 700 plus-page book and its companion CD will help you fly into game design with Allegro. It has a step-by-step approach with professional game examples. Get a sneak peak of this book by searching for the title at books.google.com.

Looking forward

Developers are hard at work, and you can expect version 5 very soon. The latest work-in-progress release reflects features that would be available in it and the library surely has undergone huge changes. First off, version 5 comes with a lot of plug-ins and add-ons bundled in one package,

like AllegroGL, GIF, JPEG, OGG, lots of drivers, a true type font plug-in, and tons more. It has updated all the internals like I/O event management systems et al, including OpenGL and DirectX drivers, new and rebuilt routines, along with many other new and advanced features. The visual difference in Allegro 4.2.2 and 5 is some change in syntax, new abstract datatypes, modified and new routines that simplify programming and service the new feature updates and optimisations. Check alleg.sourceforge.net/changes-4.9.html to have a look at the version updates and the differences.

Allegro 5 documentation is not quite impressive at this moment, as it is still under development. Allegro 4.2.2 programmers might have to face some difficulty porting their code from 4.2.2 to 5 due to the new syntax and structure. As for when Allegro 5 will be out, the answer I got at their IRC channel was: "It will be done, when it is done!"

At the end of the day

To design a good game, one has to understand game design techniques and algorithms. Advanced and complex libraries and syntax distract beginners from learning game design because of the complexities in coding, syntax, implementations and structures. Allegro's main speciality is its simplicity and the power it delivers, hiding the unnecessary and complex internal details of the system from the user.

A lot of professionals use Allegro and a number of games (including commercial paid games) have been made with Allegro. Being open source, users get a great programming library that is even used by professional programmers.

The word 'allegro' means a piece of music that is full of life and upbeat. The Allegro library brings that spirit to programming, and renders your dreams of making real games come alive with 32bpp colour and OpenGL rendering. 

REFERENCES AND LINKS

- Allegro home page: alleg.sourceforge.net
- AllegroGL home page: allegrogl.sourceforge.net/wiki/AllegroGL
- About Shawn Hargreaves: www.talula.demon.co.uk
- Community site: www.allegro.cc
- Wikipedia: en.wikipedia.org/wiki/Allegro_library
- *Game Programming All In One*, 2nd Edition, by Jonathan S. Harbour
- Freenode IRC channel: #allegro

By: Arjun Pakrashi. The author is currently studying for a B.Sc degree in computer science from Asutosh College, Calcutta University, Kolkata. His main areas of interest are open source software, Linux programming and data structures. He plans to do research-based work, and become an OSS contributor.

The Joy of Programming



S.G. GANESH

Understanding ‘typedef’ in C

The ‘typedef’ feature, though used widely, is often not understood well by programmers. In this column, we’ll look at some interesting aspects of this keyword.

Try answering these questions first; some of the answers might surprise you!

- 1 Does *typedef* mean ‘type declaration’ or ‘type definition’?
- 2 Why do we need *typedef* when we can directly declare variables with specific types?
- 3 Does *typedef* help in increasing portability of the code?
- 4 Is *typedef* a storage class (like *static*, *extern* etc.)?
- 5 Why is that languages like Java do not have *typedef* or an equivalent feature?

1 *typedef* stands for ‘type definition’; however, it’s a misnomer. *typedef* never ‘defines’ a type, it just declares a type. What does this mean?

A definition is always associated with a space allocated for it, while a declaration is just about giving information to the compiler (think of the difference between a function declaration and a definition, for example). When we use a *typedef*, say, *typedef int INT*, we declare *INT* to be of type *int*; there is no space allocated for *INT* and this *typedef* details are lost when the code gets compiled. So, *typedefs* are always declarations.

2 The main use of *typedef* is in abstracting the type details. Consider the example of *FILE** that we use for I/O—we use it without knowing anything about the underlying *struct* and that it’s a *typedef*!

```
struct _iobuf {  
    char *_ptr;  
    int _cnt;  
    char *_base;  
    int _flag;  
    int _file;  
    int _charbuf;  
    int _bufsiz;  
    char *_tmpfname;  
}; // one possible implementation  
typedef struct _iobuf FILE;
```

The detail behind *FILE* is abstracted and the user uses *FILE* freely as if it is a datatype. So *typedefs* are useful in simplifying the use of complex declarations.

3 C is a low-level language and its programs can have

many details that are implementation dependent. An important benefit in using *typedef* is that it increases the portability of programs. Consider the prototype of *strlen*: *size_t strlen(const char *)*. The *size_t* is a *typedef* of *unsigned int* or *unsigned long* depending on the platform; so we can use the program that uses *strlen* in another platform without worrying if the actual return type is *unsigned long* or *unsigned int*.

4 Yes. The grammar for C specifies:

```
storage-class-specifier : one of  
    auto register static extern typedef
```

So, *typedef* is also considered as a storage class specifier!

Initially, this question may not make any sense at all because we know that *typedef* is concerned with types and it has nothing to do with storage classes. For those who don’t know what a storage class is: a storage class specifies how/where to allocate space for the variables; for example, the *auto* storage class specifies that the variable is local to a function and should be created and destroyed as and when a function call is made and returned. What is *typedef* to do with storage classes? Storage classes such as *static*, *extern*, etc., cannot occur together; also, storage classes cannot be type-defined (a *typedef* specifies a type and not how/where it is allocated). So the creators of the language designed the grammar of the language using this insight and treated *typedef* as a storage class (as ‘syntactic convenience’).

5 C is a low-level language and, hence, for portability and abstraction, *typedefs* are very useful. Languages like Java have alternative and better abstraction facilities (we can declare a *FILE* class, for example); also, higher-level languages are often more portable than C. So, most of the higher-level languages do not need a *typedef* like feature! 

S.G. Ganesh is a research engineer in Siemens (Corporate Technology). His latest book is “60 Tips on Object Oriented Programming”, published by Tata McGraw-Hill in December last year. You can reach him at srganesh@gmail.com.

CodeSport

Welcome to another instalment of CodeSport. In this month's column we will explore a few programming puzzles requested by some of our readers.

Thanks to all the readers who sent in their solutions/ comments to the problems we discussed in last month's column. Last month's takeaway problem was to design a data structure that could support the following two operations on a set of integers, namely:

1. *Insert(S,x)* inserts x into the set S
2. *delete_half_set* deletes $\text{ceil}(S/2)$ elements from the set S

The data structure to be designed needed to be such that a sequence of M operations could run in $O(M)$ time. In other words, the amortised complexity of both *insert* and *delete_half_set* had to be constant, when a sequence of M operations consisting of *insert* and *delete_half_set* was performed on the data structure.

We can use an unsorted array, so *insert* takes $O(1)$ time. For DELETE-LARGER-HALF, we first find the median of the array and partition the array around the median. We delete the larger side of the partition. Now, if you recall the discussion we had on amortised analysis in last month's column, we use the accounting method of amortised analysis. For the amortised analysis, insert each item with two tokens on it. When you perform a DELETE-LARGER-HALF operation, each item in the list pays one token for the operation. When you delete the larger half, the tokens on these items are redistributed on the remaining items. If each item on the list starts with two tokens, then each has one after the median, and then each item in the deleted half gives its token to one of the

remaining items. Thus, there are always two tokens per item and we get constant amortised time. We see that *delete* takes place in $O(1)$ amortised time and *insert* also takes place in $O(1)$ amortised time.

Some of the readers had requested a discussion on computer science puzzles in this month's column as it would be useful in preparing for interviews. So let us take a break from algorithms and data structures (don't worry, next month we will come back to it, when we discuss number theory algorithms), and instead discuss a few popular puzzles.

Let us get started with a couple of simple ones from probability.

One of your friends (whom you have not met for a long time) tells you that he is now married and has two children. He also mentions that the older child is a boy. Now, being a probability enthusiast, he asks you to guess whether the second child is a boy or a girl. So you want to compute the probability that the second child is a girl given that the first child is a boy. There are four possible outcomes for two children, namely (Boy1, Boy2), (Boy1, Girl2), (Girl1, Boy2), (Girl1, Girl2). Since we know that the first child is a boy, the possible outcomes now are restricted to either (Boy1, Boy2) or (Boy1, Girl2). Since there are only two outcomes possible, given that the first child is a boy, the probability that the second child is a girl is half. Your friend decides to tease you a little bit more. Now he

modifies his previous statement and says that: "I have two children, and at least one of them is a boy. What is the probability that I have a daughter also?" Though it is very quick, but incorrect, to jump to the conclusion that since one child is a boy, and a boy or girl is an equally likely outcome for a child, the probability that he has a daughter is half. But this is incorrect. Let us look at the list of possible outcomes again, given that one of the children is a boy (note that now we do not know whether it is the first or second child).

Hence, we have only three valid outcomes (Boy1, Girl2), (Girl1, Boy2), and (Boy1, Boy2)—ask yourself why we did not consider the combination of (Boy2, Boy1). Out of these three possible outcomes, only the outcome (Boy1, Girl2) allows for the second child to be a girl. Hence, we have the probability of the second child being a girl as 1/3 given that the first child is a boy.

Another favourite probability puzzle is the Monty Hall problem. You are a participant in a game show. You are shown three doors by your host, who says that two of them have nothing behind them and one of them has a brand new car. He says that if you can correctly select the door behind which the car is parked, you can take home the car as your prize. Now, he asks you to choose your door. Once you have chosen your door, he opens one of the other two doors and shows that there is nothing behind it. Now he asks you whether you want to switch your choice from what you had chosen earlier. What should you do? Again the simplest answer is that now that we have only two doors that can lead to the car, the probability of either of them having the car behind it is $\frac{1}{2}$. Hence, we should stay with our previous choice. However, this answer is incorrect and I leave it to you to figure out why.

Our next question is a logic puzzle. There are three baskets. One of them has apples, one has oranges only and the other has a mixture of apples and oranges. There are three labels, with one each placed on each basket, namely 'oranges', 'apples' and 'mixture'. The name labels placed on top of all three baskets are incorrect (i.e., if the label on top of a basket says 'oranges', that basket definitely does not contain oranges). You are allowed to select only one fruit from one of the three baskets. Using this information, your job is to place the labels correctly on all the three baskets.

At first glance, the problem seems difficult. Since by picking up one fruit, you can only label that basket correctly, how can we distinguish between the other two baskets? The trick is to select the fruit from the basket labelled as 'mixture'. It can either be an apple or an orange. If it is an apple, we can label this basket correctly as 'apple'. Consider the remaining two baskets, namely the baskets labelled 'oranges' (denoted as Basket1) and 'apples' (denoted as Basket2). Since 'oranges' is the wrong label for Basket1, and we already know that Basket3 contains apples, this basket can only be the 'mixture'

basket. Since this is the 'mixture' basket and we know that Basket3 (the one originally labelled as 'mixture' from which we selected our fruit) contains apples, we can correctly label Basket2 as 'oranges'. A similar argument follows if the fruit we selected is an orange. So by selecting only one fruit, we can label all three baskets, correctly.

Let us now move on to a more programming-related question. You are given an array of N integers (there are no duplicates in the array). Consider the subsets of this set whose cardinality is (N-1). How many such sets are there? There are (N-1) such sets. For each such set, your program should output the product of the elements present in it. For instance, if you are given the array containing 10, 20, 30, 40, we have the three subsets, {10,20,30}, {20,30,40}, and {30,40,10}. The algorithm should output the three values 6,000, 24,000 and 12,000. We can write a naive algorithm that computes each of these products individually and hence will take $O(N)$ time. Instead, you are asked to come up with an $O(N)$ algorithm for computing all the (N-1) products.

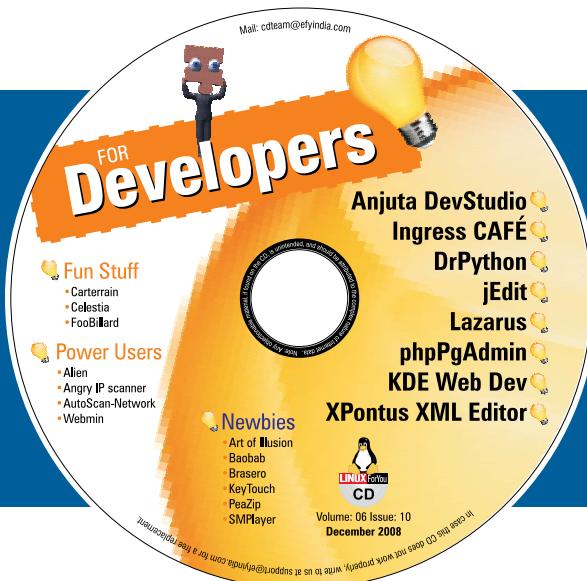
The trick is to recognise the common part in computing each of the products. Number these subsets as 1, 2 and 3, where Subset1 will not contain Element1 in Set A. Subset2 will not contain Element2 in Set A, etc. For example, we label the {20,30,40} subset as Subset1, the Subset {30,40,10} as Subset₂ and {10,20,30} subset as Subset₃. The key insight is to observe that the product of elements in each Subset_i is nothing but the product of all numbers in Array A divided by the one element in A that is missing from that particular Subset_i. We now compute the product of all elements in A, namely 'Prod_n', and then for each subset, divide 'Prod_n' by A[i] if we want to compute the product of Subset_i. This gives a $O(N)$ algorithm. I leave it to the reader to write the actual code for this.

For this month's takeaway problem, let us consider the following puzzle. You are entering a room that contains N people. What is the probability that there is someone in the room whose birthday is on the same day as yours? You can assume that there are no leap years and all years have only 365 days. Also assume that the birthday being any day of the year is equally likely.

If you have any favourite programming puzzles that you would like to discuss on this forum, please send them to me. Feel free to e-mail your solutions and feedback to me at *sandyasm_AT_yahoo_DOT_com*. Till we meet again next month, happy programming!



Sandy Mannarswamy is a specialist in compiler optimisation and works at Hewlett-Packard India. She has a number of publications and patents to her credit, and her areas of interest include virtualisation technologies and software development tools.



Start Coding!

A collection of IDEs and other tools to get you started with programming.

LFY ends the year 2008 by bundling some of the nifty IDEs and other tools that make a developer's life a bit easy. All the users out there are keeping their fingers crossed to get a sneak peak into some of the new breed of applications that may get released in 2009 maybe.

Anjuta DevStudio is a versatile IDE for C/C++ and other languages. It has been written for GTK/GNOME and features a number of advanced programming features. These include project management, application wizards, an on-board interactive debugger, and a powerful source editor with source browsing and syntax highlighting.

[/software/developers/anjuta](#)

DrPython is a highly customisable, simple, and clean editing environment for developing Python programs. It is written in Python, using wxPython as the GUI. The Python script language can be used to extend its functionality further through scripts, or plug-ins.

[/software/developers/drpython](#)

Gambas is a full-featured object language and development environment built on a BASIC interpreter. It's made up of a compiler, an interpreter, an archiver, a scripter, a development environment and many extension components.

[/software/developers/gambas](#)

Glade is a RAD tool to enable quick and easy development of user interfaces for the GTK+ toolkit and GNOME. By using *libglade*, Glade XML files can be used in numerous programming languages including C, C++, Java, Perl, Python, C#, Pike, Ruby, Haskell, Objective Caml and Scheme. Adding support for other languages is easy too.

[/software/developers/glade](#)

Ingres CAFÉ eliminates the time consuming tasks of acquiring, installing and configuring the many components developers need in a Java application development environment. CAFÉ delivers a one-click-and-code solution for Java development. It includes Eclipse IDE, Ingres database, Ingres Eclipse Data Tools Plug-in (DTP), Apache Tomcat, Hibernate, Java Server Faces Libraries, etc.

[/software/developers/ingres_cafe](#)

jEdit is a mature programmer's text editor written in Java. It consists of a built-in macro language, and an extensible plug-in architecture. It supports auto indent, and syntax highlighting for more than 130 languages.

[/software/developers/jedit](#)

KDevelop is an easy-to-use IDE to develop applications under UNIX,

MacOS, Windows, Solaris and BSD. KDevelop has a plugin-based architecture so functionality can be added, replaced and removed without altering core source code.

[/software/developers/kdevelop](#)

KDE Web Dev includes KDE-based programmer utilities to generate GUI dialogs, a Web IDE, a style sheet debugger, and a utility to search and replace strings. Those applications include Quanta Plus, Kommander, KXSL Debug, KImageMapEditor, KFileReplace and Kallery.

[/software/developers/kdewebedv](#)

Lazarus is a stable and feature rich visual programming environment for the FreePascal compiler. It includes a syntax-highlighting code editor and visual form designer, as well as a component library.

[/software/developers/lazarus](#)

MonoDevelop is a GNOME IDE primarily designed for C# and other .NET languages. It enables developers to quickly write desktop and ASP.NET Web applications on Linux and Mac OSX. MonoDevelop makes it easy for developers to port .NET applications created with Visual Studio to Linux and Mac OSX, and to maintain a single code base for all three platforms.

[/software/developers/monodevelop](#)

NetBeans is a free, open source IDE for software developers. The complete bundle consists of all the tools used to create professional desktop, enterprise, Web, and mobile applications with Java, C/C++ and Ruby. The version bundled with the *LFY* CD is the JavaSE bundle.

[/software/developers/netbeans](#)

phpPgAdmin is a fully-functional, Web-based administration utility for a PostgreSQL database server. It handles all the basic functionality as well as some advanced features such as triggers, views and functions (stored procs).

[/software/developers/phpgadmin](#)

XPontus XML Editor is a simple XML Editor oriented towards text editing. It can perform validation (DTD, XML Schema, Relax NG and Batch XML validation), XSL transformations (HTML, XML, PDF, SVG), schema/DTD generation, XML/DTD/HTML/XSL code completion, code formatting and much more.

[/software/developers/xpontus](#)

For newbies

Art of Illusion is a 3D modelling and rendering studio. It is written entirely in Java, and should be usable on any Java virtual machine that is compatible with J2SE 1.4 or later.

[/software/newbies/art_of_illusion](#)

Baobab is a C/GTK+ application to analyse disk usage in any GNOME environment. Baobab can easily scan either the whole filesystem tree, or a specific user-requested directory branch (local or remote). It also includes a complete file-search functionality and auto-detects in real-time any changes made to your home directory as far as any mounted/unmounted device is concerned.

[/software/newbies/baobab](#)

Brasero is an application to burn CDs/DVDs for the GNOME environment. It is designed to be as simple as possible and has some unique features to enable users to create their discs easily and quickly. It supports multiple backends: *cdrtools*,

growisofs and *libburn* (optional).

[/software/newbies/brasero](#)

KeyTouch is a program that allows you to easily configure the extra function keys of your keyboard. This means that you can define, for every individual function key, what to do if it is pressed.

[/software/newbies/keytouch](#)

PeaZip is used to create 7Z, ARC, BZ2, GZ, PEA, TAR, UPX and ZIP formats. It opens 79 file extensions including ACE, ARJ, CAB, DMG, ISO, LHA, RAR, UDF and many more archive types. PeaZip allows you to save archive layouts, apply powerful multiple search filters to archive content, handle multiple archives at once, export job definitions as command lines, bookmark archives and folders, etc.

[/software/newbies/peazip](#)

SMPlayer is a complete front-end for MPlayer, from basic features like playing videos, DVDs and VCDs to more advanced features like support for MPlayer filters, and more. One of the most interesting features of SMPlayer is that it remembers the settings of all files you play. So if you start to watch a movie but have to leave... don't worry; when you open that movie again it will resume at the same point you left it, and with the same settings: audio track, subtitles, volume...

[/software/newbies/smplayer](#)

For power users

Alien is a program that converts between the rpm, dpkg, stampede slp, and Slackware tgz file formats. If you want to use a package from a distribution other than the one you have installed on your system, you can use Alien to convert it to your preferred package format and install it.

[/software/powerusers/alien](#)

Angry IP Scanner (or simply *ipscan*) is an open source and cross-platform network scanner designed to be fast and simple to use. It scans IP addresses and ports. It simply pings each IP address to check if it's alive, then optionally, it

resolves its hostname, determines the MAC address, scans ports, etc. The amount of gathered data about each host can be extended with plug-ins.

[/software/powerusers/angryip_scanner](#)

AutoScan-Network is a network discovery and management application. No configuration is required to scan the network. Entire subnets can be scanned simultaneously without human intervention. It features OS detection, automatic network discovery, a nessus client, a Samba share browser, and its main goal is to print the list of connected equipments in a network.

[/software/powerusers/autoscan_network](#)

Webmin is a Web-based interface for systems administration working on UNIX. Using any browser that supports tables and forms (and Java for the file manager module), you can set up user accounts, Apache, DNS, file sharing and so on. Webmin consists of a simple Web server, and a number of CGI programs that directly update system files like */etc/inetd.conf* and */etc/passwd*. The Web server and all CGI programs are written in Perl version 5, and use no non-standard Perl modules.

[/software/powerusers/webmin](#)

Fun stuff

Carterrain is a 3D multi-player split screen, off-road racing game. Race against a bunch of friends crowded around a keyboard. It is available for Linux and Windows.

[/software/funstuff/carterrain](#)

Celestia is an application for the real-time 3D visualisation of space. It consists of a detailed model of the solar system, over one lakh stars, more than 10,000 galaxies, and an extension mechanism to add more objects. Celestia runs on Windows, Linux and Mac OS X.

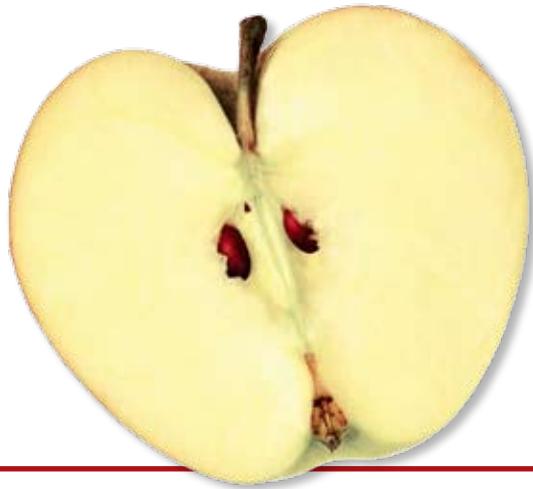
[/software/funstuff/celestia](#)

FooBillard is a free OpenGL-billiard game for Linux with realistic physics, AI-player and many game types like pool or snooker.

[/software/funstuff/foobillard](#)



A VOYAGE TO THE KERNEL



Part 7

Day 6. Segment 2.1

This article will concentrate on the computational methods used for problem solving. Unlike in the previous segment, here we will be dealing more with the theory than with trials!

A problem associated with some of the standard books on computational methods is that they require the use of proprietary software. I remember reading *Applied Quantum Mechanics* by A. F. J. Levi. The book, though very useful and interesting, provides the solutions to problems in Matlab code.

So, I shall deal with some of the free software tools that you can use while trying to apply the theory to your problems.

GNU Octave: For scientific computation

I use this wonderful software for all my work. There are few people who prefer Scilab to Octave. But I prefer the latter. There are even tools associated with some of these programs for the conversion of code from Matlab. And many of them employ simple parsers for this purpose.

Figure 1 shows a typical Octave window.

You can get the latest copy of Octave (3.0.3) from www.gnu.org/software/octave/download.html or check your distribution's software repository for it. Another option is the Qt-front-end (see Figure 2).

Some trials with Octave

To start with a simple one, let's find the square root of 3 using Octave:

```
octave-3.0.0:1> sqrt (3)
ans = 1.7321
octave-3.0.0:2>
```

Appears very simple, right?

If you wish to manipulate with matrices, you can do that in Octave. You can enter the elements of the Matrix A in the manner elucidated in Figure 3.

While building (or making) simulations, you may need to have random values for checking. You can do that by using the *rand* command followed by the number of rows and columns as shown in Figure 4.

Another fact is that you can use many of your C commands in Octave, of which the following is a simple example:

```
octave-3.0.0:9> printf ("A Voyage to Kernel\n");
A Voyage to Kernel
octave-3.0.0:10>
```

And further, Octave has many built-in, loadable and mapping functions, function files, etc, for advanced tools.

Now let's see how to save the code in Octave.

```
#!/usr/bin/octave-3.0.0 -qf
# Script written for A Voyage to Kernel
printf ("Applied Quantum Mechanics lured me into scientific
computing\n");
```

This is quite akin to the style we followed in shell programming. The first line invokes the interpreter. (Please note that if you use a different version, you need to change the interpreter name, unlike in shell programming).

Octave has many in-built mathematical conversion tools. The following code shows that

you can easily convert numbers from decimal to binary or hexadecimal:

```
octave-3.0.0:2> dec2bin (15)
ans = 1111
octave-3.0.0:4> dec2hex (475)
ans = 1DB
```

And there are many other in-built functions like `tolower`:

```
octave-3.0.0:5> tolower ("LInUX")
ans = linux
```

Another category is built-in variables (for example, `history_file`). You can get the details by issuing the corresponding command:

```
octave-3.0.0:6> history_size
ans = 1024
```

Just like in shell, you can have user-defined functions as shown below:

```
octave-3.0.0:9> function voyage_wish (wish)
>
> printf ("\a%\$\\n", wish);
>
> endfunction
octave-3.0.0:10> voyage_wish ("Happy Journey to Scientific Computation")
Happy Journey to Scientific Computation
```

Let us move on to something more complicated. If we wish to plot a function with respect to a variable by taking many parameters, it may seem a tedious task. But it is quite easy in Octave.

```
octave-3.0.0:1> t = 0:0.6:9.3;
octave-3.0.0:2> plot (t, sin(t), "3;sin(t)", t, cos(t), "+6;cos(t);")
```

You get the resultant graph as shown in Figure 5.

You can use other tools like `clearplot`, `shg`, `closeplot`, etc, for better results and to define it completely. You can also draw different types of graphs like histograms, bar graphs, pie-charts, etc, in Octave (Figure 6).

```
octave-3.0.0:4> hist (2 * t, t)
```

You can also find functions to perform computational tasks in other fields in mathematics. For example, in case we have functions like `conj (z)`, `imag (z)`, `real (z)`, etc with complex numbers.

```
octave-3.0.0:9> abs (6 + 8i)
ans = 10
```



Figure 1: A Typical Octave Window

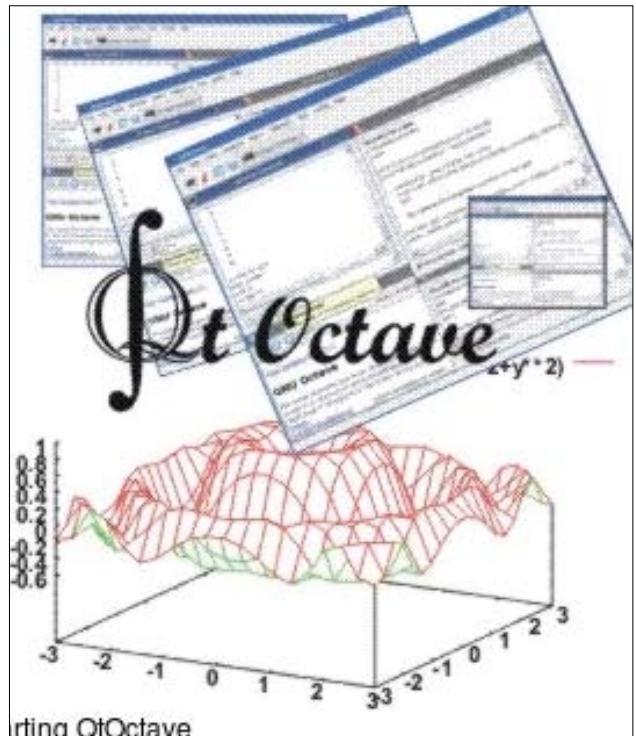


Figure 2: Qt front-end for Octave

```
octave-3.0.0:3> A = [ 3, 6; 9, 12; 15, 18 ]
A =
3   6
9  12
15 18
```

Figure 3: Elements of the Matrix A

Towards more complexity

Most of you will be familiar with the simple numerical methods that we use for computing, like the Euler and Range-kutta method. These are relatively simple yet powerful methods. For advanced-level problems we may need more functions as well. Let's take the beta function that is given mathematically as:

```
octave-3.0.0:6> b = rand (5, 5);
octave-3.0.0:7> b
b =

```

0.5649623	0.0258500	0.2716632	0.0909542	0.8134727
0.6828286	0.4537789	0.4355040	0.5319594	0.9962779
0.2223314	0.5183874	0.9763634	0.0049495	0.5672245
0.7097700	0.7305902	0.1670566	0.8416702	0.0422527
0.2323640	0.9378652	0.3217470	0.2680233	0.4316200

Figure 4: rand generates random values

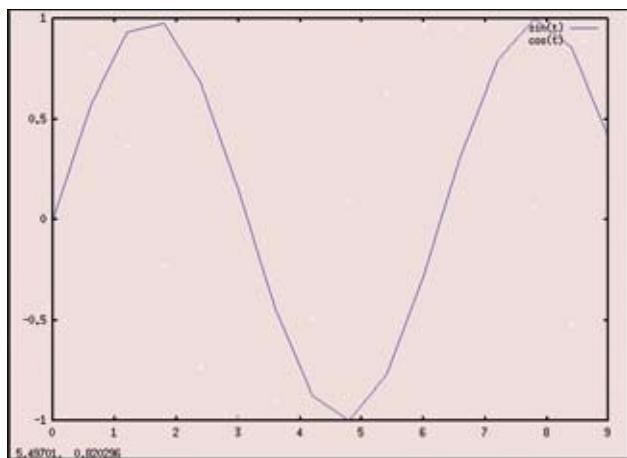


Figure 5: An Octave-generated graph

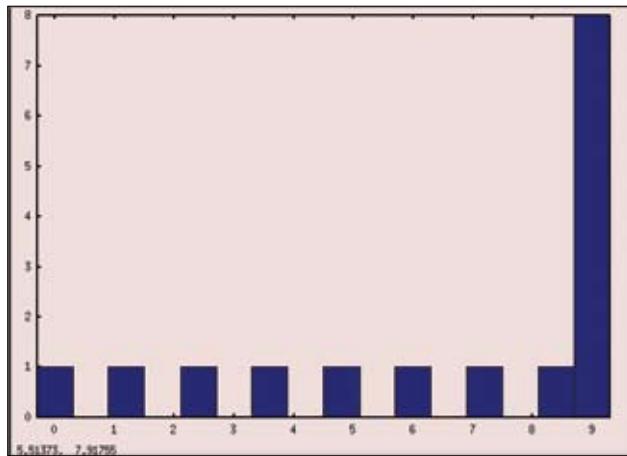


Figure 6: An Octave-generated bar graph

$$B(a,b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$$

You can straight away proceed with the 'beta' function in Octave (provided you know its use) without worrying about the stuff inside:

```
octave-3.0.0:10> beta (3,4)
ans = 0.016667
```

Things will become more complex when you need to meddle with the incomplete beta function given by:

$$\beta(a,b,x) = B(a,b) \int_0^x t^{(a-1)} (1-t)^{(b-1)} dt$$

But you are safe when you are in Octave as you have the *betainc* (x, a, b) function. So is the case with gamma and incomplete gamma functions.

$$\Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt.$$

$$\Upsilon(x, a) = \frac{\int_0^x e^{-t} t^{a-1} dt}{\Gamma(a)}$$

```
octave-3.0.0:1> gamma (3)
```

```
ans = 2
```

Hence, you can easily write algorithms (in Octave-like language) just by remembering things like *lgamma* (a, x), *gammaln* (a, x), etc.

You can make use of these types of tools while trying to meddle with tasks like finding the Hessenberg decomposition of the matrix or computing the Cholesky factor.

Computational methods

Let me try to explain the simple numerical method (in scientific computation) to deal with differential equations.

Some of you might have lost touch with all this, so I shall consider going over the concepts. Please note that the concepts developed during the early stages of the voyage into the kernel will be used for solving problems in the upcoming days.

The most important point we need to note is that in the case of differential equations, it relates a function to its derivatives, so that we can compute the function itself. Take, for example:

$$u'(t) = u(t)$$

The general solution for this will be of the form 'a constant multiplied by e^{kt} '. If you are sceptical, try differentiating the solution!

The ordinary differential equations can be represented as shown below:

$$\begin{aligned} u'(t) &= u(t), \\ u'(t) &= u^2(t), \\ u'(t) &= u(t) + \sin(t) \cos(t), \\ u''(x) + u'(x) &= x^2, \\ u'''(x) &= \sin(x). \end{aligned}$$

(The orders of the equations are different.)

We also have partial differential equations, which differ from ODEs.

And typical partial differential equations (PDE) can be classified as shown below:

$$\begin{aligned}
 u_t(x,t) &= u_{xx}(x,t) \\
 u_{tt}(x,t) &= u_{xx}(x,t), \\
 u_{xx}(x,y) + u_{yy}(x,y) &= 0, \\
 u_t(x,t) &= (k(u(u(x,t))u_x(x,t))_x, \\
 u_{tt}(x,t) &= u_{xx}(x,t) - u^3(x,t), \\
 u_t(x,t) + \frac{1}{2}u^2(x,t)_x &= u_{xx}(x,t), \\
 u_t(x,t) + (x^2 + t^2)u_x(x,t) &= 0, \\
 u_{tt}(x,t) + u_{xxxx}(x,t) &= 0.
 \end{aligned}$$

...where,

$$ut = \frac{\partial u}{\partial t}, \quad u_{xx} = \frac{\partial^2 u}{\partial x^2}$$

Out of these, there are homogeneous and non-homogeneous ones. Some equations (like Laplace equation) are homogeneous in nature:

$$u_{xx}(x,y) + u_{yy}(x,y) = 0$$

While others are non-homogeneous (like the Poisson equation):

$$u_{xx}(x,y) + u_{yy}(x,y) = f(x,y)$$

Solving problems: by computational methods

Consider the equation

$$u'(t) = f(u(t))$$

Also take:

$$u(0) = u_0$$

Applying the Taylor series for smooth functions, we get:

$$u(t + \Delta t) = u(t) + \Delta t u'(t) + \frac{1}{2}(\Delta t)^2 u''(t + \mathcal{E})$$

...provided 'u' is twice differentiable and the series is applied for $t > 0$.

Making use of the O-notation, we can have:

$$u'(t) = \frac{u(t + \Delta t) - u(t)}{\Delta t} + O(\Delta t)$$

Now we can split the time interval into small parts. Let us take t_m as an integer multiple of t . By invoking numerical approximation, we now consider v_m as the approximation of

$u(t_m)$. Let v_m be a known quantity. Then v_{m+1} can be computed. So we can write:

$$\frac{u(t_{m+1}) - u(t_m)}{\Delta t} \approx u'(t_m)$$

Using our definition:

$$\frac{u_{m+1} - u_m}{\Delta t} = f(v_m)$$

From this, v_{m+1} is evaluated. This is a simple process using the Euler method. The advantage associated is that it can be implemented on a computer for any function.

We shall deal with more complex problems and their solutions, using computational methods, in the forthcoming articles. Please note that you can directly apply some of these methods when you deal with problems in kernel programming. But for others, you may need to 'customise' the method to suit the defined problem. 

REFERENCES:

Some of the books recommended for the voyage into the new segment:

1. G. Dahlquist, A Bjorck, *Numerical Methods*, Englewood Cliffs, Prentice-Hall
2. S.D. Conte, C. de Boor, *Elementary Numerical Analysis, an Algorithmic Approach*, McGraw-Hill
3. J. D. Logan, *Applied Mathematics, A Contemporary Approach*, Wiley-Interscience
4. G. B. Whitham, *Linear and Nonlinear Waves*, Wiley-Interscience
5. H. O. Kreiss, J. Lorenz, *Initial-Boundary Value Problems and the Navier-Stokes Equations*, Academic Press
6. J. Smoller, *Shock Waves and Reaction-Diffusion Equations*, Springer-Verlag
7. W. Hackbusch: *Iterative Solution of Large Sparse Systems of Equations*, Springer Verlag
8. D. Gottlieb, S. A. Orszag, *Numerical Analysis of Spectral Methods: Theory and Applications*, Siam, Regional Conference Series in Applied Mathematics
9. E. Isaacson, H. B. Keller, *Analysis of Numerical Methods*, Wiley
10. W. Aspray, John von Neumann, *The Origins of Modern Computing*, MIT Press.

By: Aasis Vinayak PG. The author is a hacker and a free software activist who does programming in the open source domain. He is the developer and CEO of the Mozhi Search engine. His research work/publications are available at www.aasisvinayak.com



Scripts in Nautilus

Nautilus scripts add extra functionality to your file manager. Here is a script to move files from one directory to another:

```
#!/bin/bash
location=`zenity --file-selection --directory --title="Select a directory"`
for arg
do
if [ -e "$location"/"$arg" ];then
zenity --question --title="Conflict While Moving" --
text="File \"$location\"/\"$arg\" already exists. Would you like to replace it?"
case "$?" in
  1 ) exit 1 ;;
  0 ) mv "$arg" "$location" ;;
esac
else
mv "$arg" "$location"
fi
done
zenity --info --text "Finished Moving Files"
```

Save it as a file, copy it to your `.gnome2/nautilus-scripts/` directory and make it executable. Now whenever you use Nautilus, if you right click, you will see this script in the drop-down menu. You can add other scripts in a similar manner.

—Steven Seabolt papafreebird@gmail.com

Keep a check on remote logins

Generally, we use SSH to log in to another system and work on it from our terminal. If this 'other system' is yours, then I'm sure you'd be interested in knowing who all have logged in to your system remotely. You can find that out by using the following command:

```
who | grep -wv ':0'
```

The output will display the IP from which a person has logged in to your machine as well as the person's user name.

If you want to know the history of all these remote logins, use the following command:

```
last -ad | grep -wv '0.0.0.0'
```

—Rajesh Battala, rajesh.battala@gmail.com

Recover Firefox tab

What if you close an important tab in Firefox accidentally? Or if you want to hide the tab for some time. No worries... just press `Ctrl+Shift+T`. Woohooo! It's back again.

—Unmesh Jadhav, unmeshjadhav@gmail.com

Line numbers in Vim

Sometimes, to debug a long script, it helps if we can see the line numbers. To display line numbers on the left side in each shell script file, simply add the following line in your `/etc/vimrc` file:

```
set number
```

That's it! Now onwards, each file will display the line number on the left side when opened in Vim.

—Jasvendar Singh M. Chokdayat, theindianjash@gmail.com

Check network status

To check the status of a network interface, you can use the following command:

```
mii-tool
```

—Melvin Lobo, melvin.lobo@axonnetworks.com

Increase the semaphores count in a Linux machine

When you stop all services, the semaphores and shared memory segments have to be removed. If not, you will be able to see them using the *ipcs* command. You can try to manually remove them using the *ipcrm* command. For example, to remove the semaphore, execute the following commands:

```
# ipcs -a
...
--- Semaphore Arrays ---
key      semid      owner      perms      nsems
0x00000000 201293824  apache    600          1
...
# ipcrm -s 201293824
```

See *man ipcrm* for more information.

The following example shows how to increase the number of semaphores on Fedora. First get the current semaphores value:

```
# /sbin/sysctl -a | grep sem
kernel.sem = 200      32000   32      128
```

Now, we set the new value:

```
# /sbin/sysctl -w kernel.sem=250
```

Add the new value into the */etc/sysctl.conf* file in order for the change to persist across system reboots:

```
kernel.sem = 250
```

To save the changes, execute the following:

```
# sysctl -f
```

—Kiran Chand K, *kiran.virus@gmail.com*

All mp3 files in one place

The following command will find all files that end with '.mp3' and copy them to a directory in one single step:

```
find / -type f -name "*mp3" -exec cp {} DIR \;
```

...where *DIR* is the directory where you want to copy your files to. The source and destination can be any path.

—Rihaz Jerrin T.P, *rihaz.jerrin@gmail.com*

Port forwarding in Linux

You can do port forwarding in Linux using SSH as follows:

```
ssh -L localport:host:hostport user@ssh_server
```

...where the *-L* flag tells SSH to port the local port to the remote host, the parameter 'localport' is the port number, 'host' is the remote host, 'hostport' is the remote port of the remote host, 'user' is the user that has SSH access to the remote host, and 'ssh_server' is the SSH server that will be used for forwarding/tunneling.

For example:

```
ssh -L 8888:www.linuxhorizon.ro:80 user@computer
```

The advantage of using it is that you can use the proxy services of some other system with the help of port forwarding—simply give the remote server port number to the proxy settings of your local system.

—Siddharth, *siddharth3013@gmail.com*

Run GUI apps from remote hosts

You want to use a GUI application (say Firefox) not installed on your system. But it's installed on another machine in the network. You can still use the program on your system by running the following set of commands:

```
$ ssh user@serverIP -f -X xeyes
$ ssh user@serverIP -f -X xterm
$ ssh user@serverIP -f -X /opt/firefox/firefox
```

Here, *user@serverIP* is your remote login name and password. The *-f* flag requests SSH to go to background just before command execution. This is useful if SSH is going to ask for passwords or passphrases, but the user wants it in the background. It is the recommended way to start GUI programs at a remote system. The *-X* flag enables X11 forwarding.

—Remin, *remin13@gmail.com*



Share Your Linux Recipes!

The joy of using Linux is in finding ways to get around problems—take them head on, defeat them! We invite you to share your tips and tricks with us for publication in LFY so that you can reach a wider audience. Your tips could be related to administration, programming, troubleshooting or general tweaking. Submit them at <http://www.linuxforu.com>

The sender of each published tip will get an LFY T-shirt.

LINUX JOBS

Post: Sr. Linux Administrator

Company: Yahoo Software Development India Pvt. Ltd.
Profile: Candidate should have good exposure to shell/ perl scripting.
Exp.: 9-14
Location: Bangalore
Email: barrgavi@yahoo-inc.com

Post: Testing Lead

Company: Agilis Information Technologies Intl. Pvt. Ltd.
Profile: Applicant should have experience in testing of products or modules, querying the oracle database, long or short term programming experience etc.
Exp.: 4-7
Location: Delhi
Email: careers@agilisinternational.com

Post: Project Manager- Learning Technologies

Company: Sify Ltd.
Profile: Engineering Graduate with knowledge of saba product, Unix, Linux, windows platform etc.
Exp.: 8-10
Location: Chennai
Email: neha.khicha@sifycorp.com

Post: Unix Administrator

Company: Source One Management Services Pvt. Ltd.
Profile: Should possess knowledge of Unix/ Linux with experience in HP-UX environment, handling customer base environment etc.
Exp.: 6-9
Location: Bangalore
Email: amit.mishra@sourceone.co.in

Post: Oracle DBA

Company: IMSI India Pvt. Ltd.
Profile: Incumbent should possess technical knowledge of oracle versions 9i and 10g with experience of Oracle performance tuning etc.
Exp.: 4-9
Location: Bangalore
Email: chandni.suri@imsiglobal.com

Post: Sr. Oracle Apps DBA

Company: EMC Data Storage Systems (India) Pvt. Ltd.
Profile: This role requires thorough understanding of complex database architecture, strict change control process and proactive performance tuning.
Exp.: 5-10
Location: Bangalore
Email: Krishnan_Deepthi@emc.com

Post: Device Driver

Company: Cisco Systems (India) Pvt. Ltd.
Profile: BE /B.Tech /M.E /M.Tech /MCA with experience in C/ C++, Unix/ Linux, kernel, device driver, firmware etc.
Exp.: 5-10
Location: Bangalore
Email: somahara@cisco.com

Post: Sr. Oracle DBA

Company: Mavensoft Systems Pvt. Ltd.
Profile: Bachelor's Degree in computer science or equivalent with experience as a DBA, supporting HA systems etc.
Exp.: 6-8
Location: Pune
Email: careers@mavensoft.com

Post: PHP Developer

Company: Optima IT venture Pvt. Ltd.
Profile: Bachelors degree or higher in computer science or equivalent is required.
Exp.: 4-9
Location: Mumbai
Email: jobs@optimags.net

Post: Systems Engineer

Company: VMWare Software India Pvt. Ltd.
Profile: Applicant should have good communication skills with 4+ yrs of experience.
Exp.: 4-9
Location: Chennai
Email: abhyravan@vmware.com

Post: Sr. Application Administrator

Company: Edifixio India Pvt. Ltd.
Profile: Should have experience in patching, cloning, system admin tasks and trouble shooting Oracle/ SAP Applications. Excellent Linux/ Unix administration skills are mandatory.
Exp.: 5-10
Location: Kolkata
Email: Debosmita.Mukherjee@edifixio.com

Post: Unix Administrator

Company: Emerson Design Engineering Center
Profile: Good understanding of TCP/ IP and network technologies such as firewalls, switches, routers, load balancers, proxies, cache servers, etc.
Exp.: 4-9
Location: Mohali
Email: aditya.mehra@emerson.com

Post: Program Manager- Training & Development

Company: Expertus Infotech Pvt. Ltd.
Profile: Must have good experience in training & software development, training manager with software development experience.
Exp.: 5-10
Location: Bangalore
Email: prabhavathy@expertus.com

Post: Module Leader

Company: Birlasoft Ltd.
Profile: Experience of credit cards domain would be an advantage.
Exp.: 4-6
Location: Hyderabad
Email: Ramarao.kandhi@birlasoft.com

Post: Kernel Engineer

Company: Careernet Technologies Pvt. Ltd.
Profile: Strong understanding of UNIX/ Linux/ BSD internals with experience of device drivers.
Exp.: 4-9
Location: Bangalore
Email: karthik.vi@careernet.co.in

Post: Component Developer

Company: Photon Infotech Pvt. Ltd.
Profile: Applicant should have knowledge of J2EE, .Net, Linux, Unix, Windows 2003 server etc.
Exp.: 6-11
Location: Bangalore
Email: prabhu@photoninfotech.com

Post: Technical Support Engineer- Telecom Core Network

Company: Mavenir Systems
Profile: Prior Deployment/ Maintenance/ testing experience in GSM/ IMS/ CDMA/ UMTS with strong knowledge on SS7, VOIP and IMS protocols etc.
Exp.: 4-7
Location: Bangalore
Email: gudla@mavenir.com

Post: IVR- Solution Integrator

Company: Visnova Solutions Pvt. Ltd.
Profile: Incumbent should have good communication skills with 5+ yrs of experience.
Exp.: 5-9
Location: Delhi/NCR
Email: rpo@visnovasolutions.com

Post: Technical Lead

Company: Sonim Technologies (India) Pvt. Ltd.
Profile: Should have knowledge of SIP, PoC & VoIP with excellent programming skills in C++ or C# & .NET.
Exp.: 5-8
Location: Bangalore
Email: sandesh.km@sonimtech.com

Post: J2ee Lead

Company: Nuware Systems Pvt. Ltd.
Profile: Candidate should have excellent programming skills in Java-J2EE with good designing skills etc.
Exp.: 4-9
Location: Bangalore
Email: madhulatha.avisa@nuware.com

Post: Storage Administrator

Company: Source One Management Services Pvt. Ltd.
Profile: Engineering Degree from Premier Institute or Bachelor of Computer Science/ Bachelor of Science/ Equivalent Degree or Diploma in Computer Science in Electronics or any bachelors degree.
Exp.: 4-9
Location: Chennai
Email: amit.mishra@sourceone.co.in

Post: Application Support Lead

Company: Nuware Systems Pvt. Ltd.
Profile: Should have worked on giving different levels of support in various platforms like Unix, Linux and windows with knowledge of database like SQL & Oracle and also on basic java applications.
Exp.: 4-6
Location: Bangalore
Email: reshmiya@nuware.com

Post: Sr. Java/J2EE

Company: BPA Technologies Inc.
Profile: Candidate should have software development experience using standard web technologies, e.g. Java, JSP, SOAP, XML, Hibernate, JBoss, HTML, CSS, Struts. Must have strong relational database experience, Linux/ Unix experience etc.
Exp.: 5-8
Location: Visakhapatnam
Email: seshu@bpatech.com

Get the Best Paying Jobs @ naukri.com
2,00,000 Jobs to choose from!

35,000 Recruiters | Get Jobs by email | Block Your Current Employers

Powered by
naukri.com[®]

India's No.1 Job Site

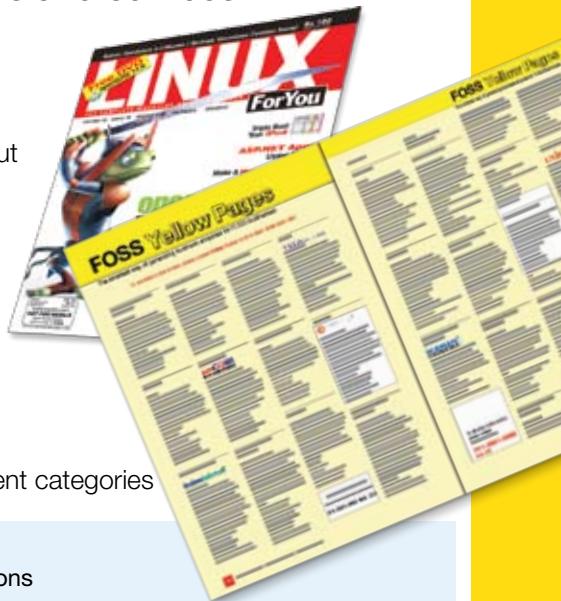
For complete recruitment solutions Employers can call us at: 0120-(3082097) or email us at vinny.ganju@naukri.com

FOSS Yellow Pages

The best place for you to buy and sell FOSS products and services

HIGHLIGHTS

- A cost-effective marketing tool
- A user-friendly format for customers to contact you
- A dedicated section with yellow back-ground, and hence will stand out
- Reaches to tech-savvy IT implementers and software developers
- 80% of LFY readers are either decision influencers or decision takers
- Discounts for listing under multiple categories
- Discounts for booking multiple issues



FEATURES

- Listing is categorised on the basis of products and services
- Complete contact details plus 30-word description of organisation
- Option to print the LOGO of the organisation too (extra cost)
- Option to change the organisation description for listings under different categories

TARIFF

Category Listing

ONE Category	Rs 2,000
TWO Categories.....	Rs 3,500
THREE Categories.....	Rs 4,750
ADDITIONAL Category	Rs 1,000

Value-add Options

LOGO-plus-Entry.....	Rs 500
Highlight Entry (white background).....	Rs 1,000
Per EXTRA word (beyond 30 words).....	Rs 50

KEY POINTS

- Above rates are per-category basis.
- Above rates are charges for publishing in a single issue of LFY.
- Max. No. of Words for Organisation Description: 30 words.

TERMS & CONDITIONS

- Fill the form (below).
- You can use multiple copies of the form for multiple listings under different categories.
- Payment to be received along with booking.

 Tear & Send

 Tear & Send

ORDER FORM

Organisation Name (70 characters): _____

Description (30 words): _____

Email: _____ Website: _____

STD Code: _____ Phone: _____ Mobile: _____

Address (will not be published): _____

City/Town: _____ Pin-code: _____

CATEGORIES

<input type="checkbox"/> CONSULTANTS	<input type="checkbox"/> HIGH PERFORMANCE COMPUTING	<input type="checkbox"/> SOFTWARE DEVELOPMENT
<input type="checkbox"/> CONSULTANT (FIRM)	<input type="checkbox"/> IT INFRASTRUCTURE SOLUTIONS	<input type="checkbox"/> TRAINING FOR PROFESSIONALS
<input type="checkbox"/> EMBEDDED SOLUTIONS	<input type="checkbox"/> LINUX-BASED WEB-HOSTING	<input type="checkbox"/> TRAINING FOR CORPORATE
<input type="checkbox"/> ENTERPRISE COMMUNICATION SOLUTIONS	<input type="checkbox"/> MOBILE SOLUTIONS	<input type="checkbox"/> THIN CLIENT SOLUTIONS

Please find enclosed a sum of Rs. _____ by DD/ MO//crossed cheque* bearing the No. _____ dt. _____ in favour of

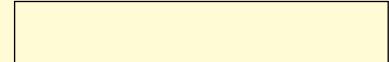
EFY Enterprises Pvt Ltd, payable at Delhi. (*Please add Rs. 50 on non-metro cheque) towards the cost of _____ FOSS Yellow Pages advertisement(s)

or charge my credit card VISA Master Card Please charge Rs. _____

against my credit card No. 

C V V No. _____ (Mandatory)

Date of Birth _____ / _____ / _____ (dd/mm/yy) Card Expiry Date _____ / _____ (mm/yy)



Signature (as on the card)



EFY Enterprises Pvt Ltd., D-87/1, Okhla Industrial Area, Phase 1, New Delhi 110 020
Ph: 011-26810601-03, Fax: 011-26817565, Email: info@efyindia.com; Website: www.efyindia.com

To Book Your Listing, Call: Dhiraj (Delhi: 09811206582), Somaiah (B'llore: 09986075717)

FOSS Yellow Pages

The best place for you to buy and sell FOSS products and services

To advertise in this section, please contact:

Dhiraj (Delhi) **09811206582**, Somaiah (Bangalore) **09986075717**

Computer (UMPC) For Linux And Windows

COMPTEK INTERNATIONAL

World's smallest computer comptek vibrain B1 UMPC with Linux, Touch Screen, 1 GB RAM 60GB, Wi-Fi, Webcam, upto 6 hour battery (opt.), USB Port, max 1600x1200 resolution, screen 4.8", 7.5"x3.25" Size, weight 526 gm.

New Delhi

Mobile: 09968756177, Fax: 011-26187551

Email: comptekdelhi@compteki.com

Web: www.compteki.com

or www.compteki.in

Enterprise Communication Solutions

Keen & Able Computers Pvt. Ltd.

Microsoft Outlook compatible open source Enterprise Groupware

Mobile push, Email Syncing of Contacts/Calendar/Tasks with mobiles

- Mail Archival
- Mail Auditing
- Instant Messaging

New Delhi

Tel: 011-30880046, 30880047

Mobile: 09810477448, 09891074905

Email: info@keenable.com

Web: www.keenable.com



Netcore Solutions Pvt. Ltd.

No.1 company for providing Linux Based Enterprise Mailing solution with around 1500+ Customer all over India. Key Solutions:

- Enterprise Mailing and Collaboration Solution
- Hosted Email Security
- Mail Archiving Solution
- Push Mail on Mobile
- Clustering Solution

Mumbai

Tel: 022-66628000

Mobile: 09322985222

Email: kalpit@netcore.co.in

Web: www.netcore.co.in

FOSS Yellow Pages

The best place for you to buy and sell FOSS products and services



Red Hat India Pvt. Ltd.

Red Hat is the world's leading open source solutions provider. Red Hat provides high-quality, affordable technology with its operating system platform, Red Hat Enterprise Linux, together with applications, management and Services Oriented Architecture (SOA) solutions, including JBoss Enterprise Middleware. Red Hat also offers support, training and consulting services to its customers worldwide.

Mumbai

Tel: 022-39878888

Email: marketing-in@redhat.com

Web: www.redhat.in

IT Infrastructure Solutions

Absolut Info Systems Pvt Ltd

Open Source Solutions Provider. Red Hat Ready Business Partner. Mail Servers/Anti-spam/GUI interface/Encryption, Clustering & Load Balancing - SAP/Oracle/Web/Thin Clients, Network and Host Monitoring, Security Consulting, Solutions, Staffing and Support.

New Delhi

Tel: +91-11-26494549

Fax: +91-11-4175 1823

Mobile: +91.9873939960

Email: sales@aisplglobal.com

Web: www.aisplglobal.com

Asset Infotech Ltd

We are an IT solution and training company with an experience of 14 years, we are ISO 9001: 2000. We are partners for RedHat, Microsoft, Oracle and all Major software companies. We expertise in legal software ans solutions.

Dehradun

Tel: 0135-2715965, Mobile: 09412052104

Email: piyush@asset.net.in

Web: www.asset.net.in

BakBone Software Inc.

BakBone Software Inc. delivers complexity-reducing data protection technologies, including award-winning Linux solutions; proven Solaris products; and application-focused Windows offerings that reliably protect MS SQL, Oracle, Exchange, MySQL and other business critical applications.

New Delhi

Tel: +91 11 40514199, Mobile 0-9810485777

Fax: +91 22 66459537

Email: sales@ingres.com; info@ingres.com

Web: www.ingres.com

Tel: 011-42235156

Email: ashish.gupta@bakbone.com

Web: www.bakbone.com

Clover Infotech Private Limited

Clover Infotech is a leading technology services and solutions provider. Our expertise lies in supporting technology products related to Application, Database, Middleware and Infrastructure. We enable our clients to optimize their business through a combination of best industry practices, standard processes and customized client engagement models. Our core services include Technology Consulting, Managed Services and Application Development Services.

Mumbai

Tel: 022-2287 0659, Fax: 022-2288 1318

Mobile: +91 99306 48405

Email: business@cloverinfotech.com

Web: www.cloverinfotech.com

Keen & Able Computers Pvt. Ltd.

Open Source Solutions Provider. Red Hat Ready Business Partner. Mail Servers/Anti-spam/GUI interface/Encryption, Clustering & Load Balancing - SAP/Oracle/Web/Thin Clients, Network and Host Monitoring, Security Consulting, Solutions, Staffing and Support.

New Delhi-110019

Tel: 011-30880046, 30880047

Mobile: 09810477448, 09891074905

Email: info@keenable.com

Web: www.keenable.com



LDS Infotech Pvt Ltd

Is the authorised partner for Red-Hat Linux, Microsoft, Adobe, Symantec, Oracle, IBM, Corel etc. Software Services Offered: •Collaborative Solutions •Network Architecture •Security Solutions •Disaster Recovery •Software Licensing •Antivirus Solutions.

Mumbai

Tel: 022-26849192

Email: sales@ldsinfotech.com

Web: www.ldsinfotech.com



Red Hat India Pvt. Ltd.

Red Hat is the world's leading open source solutions provider.

Red Hat provides high-quality, affordable technology with its operating system platform, Red Hat Enterprise Linux, together with applications, management and Services Oriented Architecture (SOA) solutions, including JBoss Enterprise Middleware. Red Hat also offers support, training and consulting services to its customers worldwide.

Mumbai

Tel: 022-39878888

Email: marketing-in@redhat.com

Web: www.redhat.in

TechnoInfotech™

A company focussed on Enterprise Solution using opensource software. Key Solutions:

- Enterprise Email Solution
- Internet Security and Access Control
- Managed Services for Email Infrastructure.

Mumbai

Tel: 022-66338900; Extn. 324

Email: sales@technoinfotech.com

Web: www.technoinfotech.com

FOSS Yellow Pages

The best place for you to buy and sell FOSS products and services

To advertise in this section, please contact:

Dhiraj (Delhi) **09811206582**, Somaiah (Bangalore) **09986075717**

Tetra Information Services Pvt Ltd

One of the leading open source providers. Our cost effective business ready solutions caters of all kind of industry verticles.

New Delhi

Tel: 011-46571313, Fax: 011-41620171

Email: sales@tetrain.com

Web: www.tetrain.com

Veeras Infotek Private Limited

An organization providing solutions in the domains of Infrastructure Integration, Information Integrity, Business Applications and Professional Services.

Chennai

Tel: 044-42210000, Fax: 28144986

Email: info@veeras.com

Web: www.veeras.com

Linux Desktop

Indserve Infotech Pvt Ltd

OpenLx Linux with Kalcutate (Financial Accounting & Inventory on Linux) offers a complete Linux Desktop for SME users. Its affordable (Rs. 500 + tax as special scheme), Friendly (Graphical UserInterface) and Secure (Virus free).

New Delhi

Tel: 011-26014670-71, Fax: 26014672

Email: info@openlx.com

Web: www.openlx.com

Software Subscriptions

Blue Chip Computers

Available Red Hat Enterprise Linux, Suse Linux Enterprise Server / Desktop, JBoss, Oracle, ARCServe Backup, AntiVirus for Linux, Verisign/ Thawte/GeoTrust SSL Certificates and many other original software licenses.

Mumbai

Tel: 022-25001812, Mobile: 09821097238

E-mail: bluechip@vsnl.com

Web: www.bluechip-india.com

Want to register
your organisation in
FOSS Yellow Pages

For **FREE***

CALL: DHIRAJ (DELHI) 09811206582

SOMAIAH (BANGALORE) 09986075717

*Offer for limited period.

Software Development

Carizen Software (P) Ltd.

Carizen's flagship product is Rainmail Intranet Server, a complete integrated software product consisting modules like mail sever, proxy server, gateway anti-virus scanner, anti-spam, groupware, bandwidth aggregator & manager, firewall, chat server and fax server. Infrastructure.

Chennai

Tel: 044-24958222, 8228, 9296

Email: info@carizen.com

Web: www.carizen.com



InfoAxon Technologies Ltd.

InfoAxon designs, develops and supports enterprise solutions stacks leveraging open standards and open source technologies. InfoAxon's focus areas are Business Intelligence, CRM, Content & Knowledge Management and e-Learning.

Noida

Tel: 0120-4350040, Mobile: 09810425760

Email: sales@infoaxon.com

Web: www.opensource.infoaxon.com

Thin Client Solutions

Enjay Network Solutions

Gujarat based ThinClient Solution Provider. Providing Small Size ThinClient PCs & a Full Featured ThinClient OS to perfectly suite needs of different working environment. Active Dealer Channel all over India.

Gujarat

Tel: 0260-3203400, 3241732, 3251732,

Mobile: 09377107650, 09898007650

Email: info@enjayworld.com

Web: www.enjayworld.com

Training for Corporate

Complete Open Source Solutions

RHCT, RHCE and RHCSS training.

Hyderabad

Tel: 040-66773365, 9849742065

Email: nayak.sujeet@gmail.com

Web: www.cossindia.com

To advertise in this section, please contact

Dhiraj (Delhi) **09811206582**

Somaiah (Bangalore) **09986075717**

ElectroMech

Redhat Linux and open source solution , RHCE, RHCSS training and exam center,Ahmedabad and Vadodara

Ahmedabad

Tel: 079-40027898

Email: electromech@electromech.info

Web: www.electromech.info



FOSTER Linux

Linux & Open Source Training Institute, All trainings provided by experienced experts & System Administrators only, RHCE, RHCSS, (Red Hat Training & Examination Partners), PHP, Perl, OpenOffice, Clustering, Mail Servers, Bridging the GAP by providing: Quality training (corporate & individual), Quality Manpower, Staffing and Support & 100% Placement Assistance.

Gurgaon

Tel: 0124-4268187, 4080880

Mobile: 09350640169, 09818478555

Email: info@fosteringlinux.com

Web: www.fl.keenable.com

Gujarat Infotech Ltd

GIL is a IT company and 17 years of experience in computer training field. We have experience and certified faculty for the open Source courses like Redhat, Ubuntoo, and PHP, Mysql Ahmedabad

Tel: 079-27452276, Fax: 27414250

Email: info@gujaratinfotech.com

Web: www.gujaratinfotech.com

Lynus Academy Pvt. Ltd.

India's premier Linux and OSS training institute.

Chennai

Tel: 044-42171278, 9840880558

Email: contactus@lynusacademy.com

Web: www.lynusacademy.com

Linux Learning Centre Private Limited

Pioneers in training on Linux technologies.

Bangalore

Tel: 080-22428538, 26600839

Email: info@linuxlearningcentre.com

Web: www.linuxlearningcentre.com

Netweb Technologies

Simplified and scalable storage solutions.

Bangalore

Tel: 080-41146565, 32719516

Email: info@netwebindia.com

Web: www.netwebindia.com

New Horizons India Ltd.

New Horizons India Ltd., a joint venture of New Horizons Worldwide, Inc. (NASDAQ: NEWH) and the Shriram group, is an Indian company operational since 2002 with a global footprint engaged in the business of knowledge delivery through acquiring, creating, developing, managing, lending and licensing knowledge in the areas of IT, Applied Learning, Technology Services and Supplementary Education. The company has pan India presence with 15 offices and employs 750 people.

New Delhi

Tel: 011-43612400

Email: info@nhindia.com

Web: www.nhindia.com

TNS Institute of Information Technology Pvt Ltd

Join RedHat training and get 100% job guarantee. World's most respected Linux certification. After RedHat training, you are ready to join as a Linux Administrator or Network Engineer.

New Delhi

Tel: 011-3085100, Fax: 30851103

Email: nks@tiit.co.in

Web: www.tiit.co.in

Training for Professional

IPSR Solutions Ltd

Earn RHCE / RHCSS certification, in Kerala along with a boarding & free accommodation. IPSR conducted more than 2000 RHCE exams with 95-100% pass rate. Our faculty panel consists of 15 Red Hat Certified Engineers.

Kochi, Kerala

Tel: +91 9447294635

Email: training@ipsrsolutions.com

Web: www.ipsr.org

Q-SOFT Systems & Solutions Pvt Ltd

Q-SOFT is in a unique position for providing technical training required to become a Linux Administration under one roof. Since inception, the commitment of Q-SOFT towards training is outstanding. We Train on Sun Solaris, Suse Linux & Redhat Linux.

Bangalore

Tel: 080-26639207, 26544135, 22440507

Mobile: +91 9945 282834

E-Mail: counsellors@qsoftindia.com

Web: www.qsoftindia.com

INDIA'S NO.1 SELLING CRICKET MAGAZINE !!



Lifeline of Cricket

Cricket today

www.crickettoday.in

NOW AT YOUR DOORSTEP!

CATCH ALL THE ACTION, RECORDS, REPORTS, STATISTICS, EXCLUSIVE COLUMNS

& INTERVIEWS ..JAZZED UP WITH EMOTION & AGGRESSION...

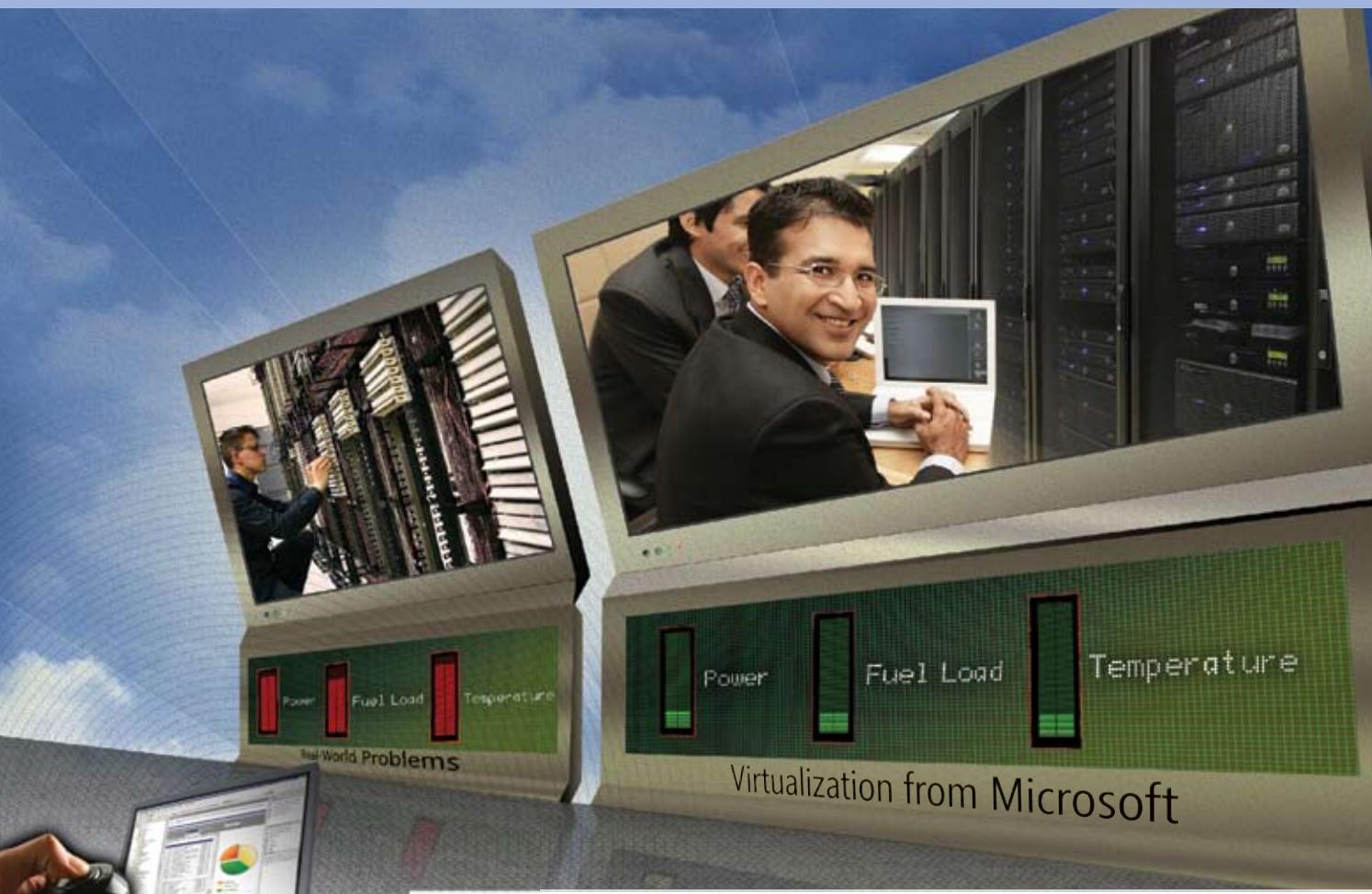
ON & OFF THE PITCH...FROM ROUND THE WORLD....



Lifeline of Cricket

Cricket today

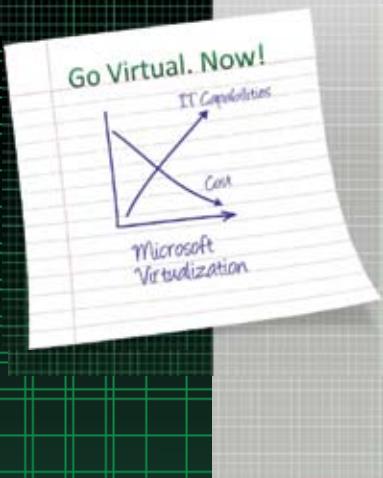
www.crickettoday.in



Accelerate your IT Capabilities. Put Brakes on Cost.

Welcome to Microsoft Virtualization.

Microsoft Virtualization breaks down barriers to creating the Virtual Enterprise. With end-to-end solutions, not only can you manage your technology infrastructure in an easy, smart and flexible manner, but also accelerate IT capabilities while reducing costs.



© 2008 Microsoft Corporation. All rights reserved. Microsoft and 'Your potential. Our Passion' are all registered trademarks of Microsoft Corporation in the United States and/or other countries.

Microsoft | Virtualization

INTEROP®

MUMBAI | 4-6 FEBRUARY, 2009

MMRDA GROUNDS, BANDRA KURLA COMPLEX

THE LEADING BUSINESS TECHNOLOGY EVENT

LAS VEGAS • MUMBAI • NEW YORK • SÃO PAULO • TOKYO

CALL FOR
PAPERS
OPEN

Learn about all the technologies that drive your business:

- Data Center
- Enterprise 2.0
- Wireless and Mobility
- Green IT
- Software-as-a-Service
- Cloud Computing
- SOA
- Open Source
- Document Management
- Application Delivery
- Security
- Storage
- Unified Communications
- Virtualization
- IP Telephony
- Telepresence
- Networking and Services
- Business Intelligence



For exhibiting and sponsorship information contact
Pankaj Jain at pankajj@ubmindia.com or call
(Mobile) + 91 98101 72077 or (Tel) + 91 11 2376 5553

Registration will open in November. To join the mailing
list, visit www.interop.com/mumbai/list

THE LEADING GLOBAL BUSINESS TECHNOLOGY EVENT IS COMING TO INDIA

Interop drives the growth and innovation of the global business technology market. As IT's most comprehensive event, Interop brings together the largest audience of enterprise buyers, SMB buyers, channel buyers, press and analysts—all in one place—to learn about the latest technologies, products and services.

Business and IT professionals attend Interop to get the information they need to purchase the latest technology. Tomorrow's market leaders are defined at Interop. Be part of the excitement this February.

